Access Control Model for WebServices eGovernment Infrastructure

TAM Ka Wing Matthew

Supervised by WONG Kam-Fai

A THESIS
SUBMITTED TO THE DEPARTMENT OF SYSTEMS ENGINEERING AND
ENGINEERING MANAGEMENT
THE CHINESE UNIVERSITY OF HONG KONG
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF MASTER OF PHILOSOPHY

This thesis is dedicated to

ALL FRONTLINE MEDICAL STAFF OF HONG KONG

They have demonstrated a high level of professionalism, dedication and devotion to their duties in the battle against Severe Acute Respiratory Syndrome (SARS). This small piece of work is dedicated to them.

Acknowledgement

First of all, I would like to thank my supervisor, Professor Kam-Fai Wong, whose rich knowledge and experience in research has certainly helped me a lot. Many times during my period of research, Profession Wong has provided valuable direction and guidelines to my work. I am also indebted to Prof Chung-Hung Cheng who has helped to review my earlier paper on eGovernment webservice infrastructure models which is the foundation of this thesis.

Thanks are also due to the thesis committee members, including Prof Chung-Hung Cheng and Prof Christopher Yang for reviewing this document.

Abstract

eGovernment is an exciting area for applying Information and Communication Technologies (ICT). ICT can improve the efficiency and effectiveness in the provision and delivery of citizen services. A critical issue for the eGovernment implementation is the interoperation problem among heterogeneous legacy government systems. In this aspect, the universal system interoperability supported by the XML-based webservices technologies can be a useful component in a holistic eGovernment infrastructure. A key requirement of the eGovernment systems is the establishment and the implementation of the right access policy to the government resources. This in turn requires an appropriate mechanism to specify the access rules. Due to the nature of webservice and the specific requirements in the eGovernment context, we propose that a more powerful and flexible mechanism is required to express the access policy more effectively in a webservices eGovernment infrastructure.

摘要

電子政府是資訊及通訊科技(ICT)的一個重要應用項目。ICT 可以幫助政府更有效及更有效率地給市民提供各種政府服務。推行電子政府的其中一個關鍵議題，是不同的原有政府系統之間的互用性問題。 針對這個議題，以 XML 爲基礎的網絡服務互用標準給電子政府提供了一個有用的技術，足以應付政府系統之間的整合需要。對於電子政府系統，提供和建立一個正確的存取政策以保障政府系統上的資源不會被擅用尤爲重要。這進一步要求一個適當的結構用以建立所須的存取規則。針對網絡服務的本質和電子政府的特定需求，本論文提出一個功能較強及較靈活的機制，用以有效地建立可應用於電子政府網絡服務系統存取規則。

Table of Contents

Acknowledgement i
Abstract i
Table of Contents iii
1 Introduction 1
2 eGovernment Requirements 5
2.1 Efficient Operation 5
2.2 Citizen-centric Government 7
2.3 Security 10
2.3.1 Confidentiality 10
2.3.2 Integrity 12
2.3.3 Availability 12
2.4 Support of eBusiness 13
3 Webservices for e-Government - A Marriage for Interoperability 15
4 A Webservices Based eGovernment Framework 17
4.1 System Component Model 17
4.2 System Access Model 18
4.3 Security Model 20
4.3.1 Access Right Model 20
4.3.2 Authentication Model 22
4.4 Transaction Model 23
5 eGovernment WebServices Access Control 25
5.1 eGovernment WebService 25
5.2 Request of Access 27
5.3 eGovernment Access Policy 30
5.3.1 Administration Based Policy 32
5.3.2 Legislation Based Policy 33
6 Research in Access Control 38
6.1 Traditional Model 38
6.2 More Advanced Models 39
6.2.1 Role-Based Access Control Model 39
6.2.2 Task-Based Authorisation Control Models 41
6.2.3 Digital Library Authorisation Model 42
6.3 Recent Works 44
6.4 Limitations of the Models 45
7 Proposed Approach 47
7.1 WebService Specific Access Control 48
7.1.1 WebService Access Rules 48
7.1.2 Authorisation Conflict Resolution 50
7.2 Subject Based Access Control 52
7.2.1 Subject Category 52
7.2.2 Subject Access Rules 53
7.2.3 WebService Registration 55
7.2.4 Authorisation Conflict Resolution 56
7.3 The WebServices 57
7.4 Combining Two Level Access Control 57
7.5 Application to Chained WebService Request 58
7.6 Comparison with the Existing Access Control Models 59
8 An Implementation Reference Model 60

8.1 Some Practical Issues 60
8.1.1 Citizen Privacy 60
8.1.2 Trust between eGovernment Systems 61
8.1.3 Authentication 62
8.2 System Architecture 64
8.2.1 eGovernment WebServices Gateway 65
8.2.2 Authentication Engine 66
8.2.3 Access Control Database 66
8.2.4 Access Control Decision Engine 67
8.2.5 A Working Scenario 67
8.3 Implementation 69
9 Evaluation of the Proposed Mechanism 74
9.1 Application Scenarios 75
9.1.1 Citizen Level Access Right 75
9.1.2 Access Means Based Authorisation 76
9.1.3 Access Right Based on Combination of User and Consumer Identity 77
9.1.4 Legislation Based Access Right 78
9.1.5 Joined-up Government 79
10 Conclusion and Future Directions 81
References 84

1 Introduction

A glance at the economic statistics of a modern country reveals that a substantial proportion of her GDP is being spent to provide the government services. As a result, it is not surprised that a government is always looking for ways to make better use of the public resource. To this end, information technologies that have been successful in improving the efficiency and effectiveness of the private sector have long been applied in governments. At first, the use was limited to the application of office automation tools like word processors and spreadsheet programs to improve the efficiency of individual tasks. Later on, sophisticated information systems were built one after the other to achieve automation on a system scale. In many modern countries, this has resulted in digital governments with a collection of heterogeneous government information systems which exchange a high volume of information with the private sector (citizens) [Schorr & Stolfo 1997].

The recent rapid development of eCommerce has demonstrated how the advancement in data communication technology (in particular the explosion of Internet connections) can enable business transactions to be conducted electronically and in a more efficient way than the traditional business process. Given the successful eCommerce implementation in the private sector, we naturally develop the same expectations for applying data communication technology in government systems. eGovernment projects may be viewed as the answers to meet these expectations.

In a narrow sense, eGovernment may be defined as simply the presence of government websites. However, to fully exploit the potential of the application of communication technology to improve the delivery of government services, the web-centric definition would seem to be too restrictive. Indeed, [Jackson & Curthoys 2001] has pointed out that eGovernment is a transformative phenomenon that based on back-office integration and shared information resources. We believe that eGovernment projects can cover:

- backoffice system integration (government reengineering and back-office process streamlining)
- integration of systems between different government bodies (G2G streamlining)
- provision of government information and services on the web (G2C & G2B application)
- provision of access to government information and services from electronic systems other than the web, eg a business's backend system (G2B application)

As pointed out in Arcieri's article [Arcieri et al 2001], a critical issue for the eGovernment projects is the interoperability problem among heterogeneous legacy government systems. There are many research efforts to address the issue. For example, Bouguettaya et al [Bouguettaya et al 2001] used metadata to assist the navigation among government databases; the Energy Data Collection (EDC) project of the Digital Government Research Centre [Ambite et 2001] applied information integration technique to provide a single integrated user interface to access large diversified information. While these efforts alleviate the problem by providing intelligence to hide the underlying heterogeneity of the databases from the user, a more general solution to the problem requires effective tools and methodologies to provide easy and seamless connections between systems that were developed by different people, running in different environments and under different software/hardware platforms. Although there have been a whole family of distributed computing solutions developed by different software vendors and standard bodies, none has received sufficient acceptance to ensure that eGovernment systems building upon which can achieve universal interoperability.

In this aspect, the webservices technology is probable a breakthrough in the area of system integration. Building upon XML, which is a truly platform-neutral technology, the webservices interoperability standards are quickly accepted by the major software industry vendors. For the first time in the software history, we now have the core technologies that promise to achieve universal software interoperability on the Internet. As a tool to solve the interoperability problem, it is not difficult to see that the Web-Services technologies can serve as a useful component in eGovernment.

Since the expectations on and the nature of the organisations in the public sector and private sector are different, there are different requirements on eGovernment projects as compared with the eCommerce projects. As such, application of webservices in the eGovernment environment should merit a study on its own. Nevertheless, to the best of our knowledge, the application of the webservices technologies in an eGovernment context is a subject still largely untouched. It is the primary objective of this paper to explore how we can incorporate the webservices models in the eGovernment infrastructure.

Due to the dependency of citizen on government services, eGovernment must be highly trustworthy. According to an eGovernment survey conducted by Taylor Nelson Sofres [Taylor Nelson Sofres 2001], security is a major concern of most potential users of the eGovernment services. We believe that the enforcement of the appropriate access policy is an essential prerequisite of any eGovernment solution, including the webservices based one. We have examined the detailed requirements on access control for a webservices based eGovernment based on our models. Based on our analysis, we conclude that the traditional access control mechanisms are inadequate for the webservice based eGovernment environment. To address this predicament, we propose a flexible framework which is composed of two key components: the webservice specific access control and the subject based control.

In what follows, we will first review the key eGovernment requirements and the importance of system interoperability in each of them (Section 2). We then examine how the webservice technologies as an effective tool for system integration can help to resolve the eGovernment problems (Section 3). Based on this background, we provide an outline of a webservice based eGovernment framework in the form of a few eGovernment webservices models (Section 4).

Having established a framework to apply the webservices technologies in the eGovernment context, we continue to examine how the security and in particular the system access control should be enforced in such an environment. In Section 5, we examine the specific access control requirements in a webservices based eGovernment environment based on our eGovernment webservices models. Section 6 is a survey on the historical researches on access control and provides an overview of how the general access control problem is handled. Our proposed model to tackle the webservices based eGovernment access control requirements is then provided in Section 7, followed by a corresponding implementation reference model and the description of a pilot implementation in Section 8. Section 9 is an evaluation of the proposed model. The last section of this paper concludes our study and provides some possible directions for further research in the area.

2 eGovernment Requirements

There are a number of structural difference between the public sector and the private one [Traunmuller & Wimmer 2000]:

- The legal framework plays a vital part in the government systems.
- There is a high level of non-instrumental rationality in the public sector.
- Usually there are many stakeholders involved in a government process, which requires high level of collaboration.

In addition, there are also different expectations to the government bodies as compared to the private sector. In particular, Government is dealing with the well being of a society as a whole and there should be equal access of all its services to all citizens [Traunmuller & Wimmer 2000]. This provides a strong justification for the implementations of eGovernment-on-the-web projects.

We will examine the implication of these differences on the eGovernment requirements in the following four areas:

- efficient operation;
- citizen-centric government;
- security;
- support of eBusiness.

2.1 Efficient Operation

With the large magnitude of government spending in term of GDP [OECD 2002], all the modern governments have been under pressure to become more efficient. For example, the "National Performance Review" (NPR) initiative team was set up in the United State to create a government "that works better and costs less" [Gore 1993]. Moreover, the advance of information technology also raises the service expectation from the citizens who will no longer be willing to accept long response time to get their service requests processed. Reduction of the processing cost and time of government services is thus an important objective of eGovernment.

At the very least, eGovernment or digital government can enhance the efficiency of the public sector by automating of labour intensive government processes. However, a more significant efficiency and productivity gain is usually achievable when a fundamental redesign of the processes is done together with a new system, effectively a reengineering of the government using information technology [Mechling 1994]. This may also involve backoffice integration within the government body to eliminate the manual transmittal of data between system components.

Both the process reengineering and backoffice integration stops at the boundary of the individual government departments. However, there are large number of government processes which require collaboration among different departments, in a way similar to the "interorganisational workflow" [Wil 2000] in the private sector. Hence, this offers ample opportunities for another level of streamlining and process reengineering beyond a single government body.

The recent SARS disease outbreak in Hong Kong does offer a good example to demonstrate the important of information sharing across the government departments. The task to fight against the disease involved different tasks including the hospital treatment for the infected, the trace of infection source, the quarantine of the potential infected, public education and the release of latest statistics of the disease to the public. Each of these tasks may be the responsibility of one government department or jointly by several departments. The effort of each department must however be well coordinated to achieve the desired result. For example, the trace of infection source was based on the information of patient obtained from the hospital treatment and any new reported SARS case should initiate a trace. In this case, it would be very useful if the systems of the responsible departments can exchange the information automatically to fit in a workflow. As another example, during the critical period, the government of Hong Kong was expected to release the latest statistics of the disease to the public every day. This required the collection of the data from the individual hospitals receiving SARS patients. The ability to extract the data from the systems of the hospital directly would make it possible to automate the entire task and certainly improve the efficiency of the reporting significantly.

In addition, as observed by Virili [Virili 2001] a large source of government inefficiency exists simply because the database systems of government branches are implemented independently. As a result, citizen data is distributed and often duplicated among these databases and there is no easy way to get a consistent view of the citizen's data across the departments.

Without the technology to achieve interoperation between the systems, inter-department or G2G (Government-To-Government) streamlining is difficult to achieve. We can thus see that webservices as an integration tool will have great value in facilitating the streamlining of G2G processes. To a less extent, the technology is also valuable in the intradepartment process reengineering exercise as a tool for backoffice integration.

2.2 Citizen-centric Government

Unlike a private organisation, one of the government's missions is to make the services available to *all* citizens. To meet this goal, eGovernment systems are required to provide equal accessibility to all the citizens in an easy-to-use way. Putting all the government services on the web is a major step toward this end, which provides around-the-clock access to the government services in any location via the Internet. This can be achieved in different stages [The Economist 2000]:

- Stage 1 – using the web to post information about various departments and agencies.
- Stage 2 – providing a two-way communication, allowing citizens to provide new information about themselves—such as a change of address—instead of telephoning or writing.
- Stage 3 – allowing a formal, quantifiable exchange of value to take place. It might be renewing a licence, paying a fine or enrolling in an educational course.
- Stage 4 – a portal that integrates the complete range of government services, and provides a path to them that is based on need and function, not on department or agency. A single log-on with password protection allows a user to get in touch with any part of government.

In the last stage, a portal is a more preferable interface to use the government services. Usually the portal provides grouping of government services by "life events": few people visit the government sites for information surfing and citizens usually use eGovernment for a purpose; the "life events" grouping provides a means for a citizen to locate the required service quickly.

It is clear that each subsequent stage requires integration of more legacy functions with the web server in a more sophisticated way and the use of webservices technologies should be able to simplify the work.

The access of eGovernment using web browser is a major improvement to government service delivery. The expectation will however be for more means to access the eGovernment. Sooner or later, following a similar access pattern in eCommerce services, the requirement to access the eGovernment over other electronic devices like smart mobile phones and PDAs will be taken for granted. In particular, service delivery to non-computing device like the touch-toned telephone handset is viewed as a way to narrow the digital divide, so that access to electronic government service is not limited to those PC-literates. Without the webservices technologies to achieve universal interoperability, the connection of the legacy government systems to all these different types of devices can be a daunting task requiring tedious case-by-case integration efforts.

The term "joined-up government" is sometimes used to describe a holistic government structure, in which the citizens requiring government services does not need to care about which department does what. One example of such vision can be found in the Italian e-Government action plan [Virili 2001]:

- Any authorised "front office" administration should supply any service to any entitled citizen;
- After being identified, the citizen should not be requested to furnish any personal information already in procession by another administration;
- Citizens should communicate any changes of their personal information only once.

For a truly citizen-centric eGovernment, the joined-up government should be an important goal to achieve. It is easy to see that the achievement of joined-up government would require high level and seamless integration among the government systems.

2.3 Security

Due to the dependency of citizen on government services, eGovernment is expected to be highly trustworthy. This demands strong security built in the eGovernment systems. Specifically, the requirements are:

- confidentiality – the government resources (information and services) can only be accessible (retrieval and update) to authorised personnel;
- integrity – the eGovernment information/services provided must be complete and accurate;
- availability – the eGovernment services should be available to the citizens at time of need and must be free from unscheduled down time.

Some may argue that there is a fourth requirement – there must be adequate audit trail of all activities accessing the government resources. We however view this need for traceability just as a means to achieve the confidential, integrity and availability requirements and thus will not treat it as a separate requirement.

2.3.1 Confidentiality

The development of digital government inevitably puts most of the citizen's personal data available in digital form, in one or more government system databases. eGovernment is about the sharing of information resources and the delivery of government information services over the Internet. Without an adequate protection for access of the eGovernment services, we will run the risk of uncontrolled exposure of sensitive citizen data and violating the data privacy requirements.

While there have been a lot of technical security solutions available, the issue of eGovernment security is more a problem to define the appropriate access policy. This is a non-trivial task that should take into consideration a number of non-technical factors like trust, legal issues, privacy, etc [Wimmer & Bredow 2001].

In general, the eGovernment access policy should achieve the protection of citizens' personal information privacy [Henderson & Snyder 1999]. This can be a very tricky requirement however as there may be an inherent conflict with the other eGovernment expectations, namely:

- personalisation of eGovernment services;
- integrated eGovernment.

To achieve personalisation of services, ie to give a customised experience of the eGovernment to each user that is appropriate to the individual, will necessitate the profiling of the user. The will however allow the system to identify the person in each access. For those who are concerned with data privacy, the possibility for a database to be built up on the access pattern of each citizen which can be used for data surveillance is simply unacceptable.

We can thus see that the challenge is to have appropriate models for user identification and authentication. Specifically, users should be provided with a range of choices over how much data they are willing to provide, in exchange to a more powerful and efficient service. In one extreme, the user can decide to use a common global digital ID to achieve a one-stop, single logon access to all eGovernment services, accepting that there will be a larger exposure of his/her access pattern. At the other end of the spectrum, anonymous access should be allowed as far as possible, and user authentication based on a separate digital identity for each application should be supported.

The implementation of an integrated eGovernment is another area which may conflict with the citizen's data privacy requirement, as the interconnection of government systems could allow free information flow from one government department to the other. In some situations, there could be specific statutory provisions for inter-department information access, to facilitate the collaboration of government bodies in specific areas. Actually, government reengineering exercises may often encounter areas which would require new legislation of this type to remove legal barriers for inter-department process streamlining. In situations where the statutory provision is not present, the share of information would require the endorsement of the data owner and the streamlined and integrated version of services can only be provided to users who accept the disclosure. We thus see that the legal framework plays a very important role to resolve this type of access requirement conflicts.

2.3.2 Integrity

Government services are required for the well-beings of citizens both individually and as a whole and the services provided are important and sometimes critical to the users. System errors in eGovernment can be costly and even have damaging effect to the society. It can thus be argued that the need for integrity in eGovernment systems is much greater compared with most of the private sector systems. As a basic requirement, we need to ensure that the information flow in the eGovernment transactions is complete and accurate.

2.3.3 Availability

To achieve true security to access government systems, we need to protect against unintended interruption of the eGovernment services, whether caused by accident or as a result of malicious attacks. This is very important especially for those government services related to the public security. Indeed, the importance of the government systems availability has received increasing attention recently when people start to think about the possible fatal consequences of breakdown in a life maintaining service (eg the firefighting service) during a critical moment.

Much of the means to enhance eGovernment availability will have to do with the overall system infrastructure, eg a more resilient data network or more redundancy incorporated in the processing equipment. However, in terms of eGovernment access the expectation will be that a wider range of channels should be supported to provide mutual backup of access. In particular, wireless access to eGovernment systems will become more important to ensure the availability of essential services during crisis situations. This will require the connection of government systems not only to the web browser, but also to different types of electronic devices that are capable to utilise the service, eg smart mobile phone, PDA device, etc.

2.4 Support of eBusiness

Today few governments would allow the economy to run entirely on its own. In fact, all the modern governments have economic policies to safeguard the economic well-being of society. As one of these economic goals, promotion of IT adoption is pursed by most governments as an opportunity to improve the living standard of the citizens and enhance the competitiveness of the society's economic power. For example, the Hong Kong SAR government has put "promoting the wider use of IT in the community and improving public access to online services" as part of the "Digital 21" Strategy [HKSAR ITSD 2001].

To this end, the government has at least two roles to play. Firstly, as the eBusiness facilitator the government is expected to ensure that the necessary infrastructure and environment is in place for electronic transactions to take place efficiently and securely. In particular, as the government is the provider of large number of services essential to the operation of every business, the efficient delivery of the eGovernment services is an important driver to the adoption of eCommerce. One way this can be done is to provide an easy mean to achieve G2B system integration to allow straight through consumption of eGovernment services.

Secondly, as a major player in the economy, the government can play the role as a leader in adoption of new technologies. Being both an effective tool for Enterprise Application Integration and an inexpensive alternative to the EDI solutions to support exchange of information between businesses (B2B integration), the webservices technologies have a lot of applications in the private sector. In this aspect, successful applications of webservices technologies in eGovernment projects can set good examples on how the technologies can be used beneficially.

3 Webservices for e-Government - A Marriage for Interoperability

The eGovernment systems are not the only applications that require integration and interoperability. Most organisations are facing with the same problem of integrating large number of legacy systems and solutions for Enterprise Application Integration (EAI) have always been in strong demand. Driven by the market, a whole family of distributed computing solutions have been invented over the years. This includes RPC from Sun, COM/DCOM from Microsoft and COBRA from the Object Management Group (OMG). Unfortunately, none of these products has enjoyed widespread acceptance and this has greatly limited the usefulness of these solutions for universal interoperability. As a result, integration solution providers are often forced to build new adapters when a new connection is required, effectively reinventing the wheel every time.

The lesson learnt is that to achieve universal interoperability, we need standards that are both vendor neutral and supported by all vendors. With the recent emergence of XML, the industry has been quick to recognise its potential as the building block for truly language and platform neutral standards. To this end, a suite of eXtensible Markup Language (XML)-based webservices technologies [Graham et al 2001] [Chappell & Jewell 2002] has been evolved as the most promising solution to ***enable*** global system interoperation:

- Simple Object Access Protocol (SOAP) is an XML-based protocol for exchanging structured information in a decentralized, distributed environment. SOAP provides a "http-friendly" protocol for systems to talk over the Internet, to request and to serve for webservices [W3C 2000];

- Web Service Description Language (WSDL) is an XML format for describing network services. WSDL allows the client to understand how to invoke a webservice [W3C 2001];
- Universal Description, Discovery, and Integration (UDDI) is a webservices based standard for registration and search of network services. UDDI provides a webservices discovery platform on the Internet [Boubez 2002].

As we can see from the definitions of webservices given by the major players (IBM [Feller], Microsoft [MSDN] [MSDN - BizTalk] and Sun [Sun 2002]), webservices are about providing services over the network via standard XML wire format. Thus, each vendor can have a different webservices architecture which is the most appropriate to its proprietary platform [Myerson 2002], and at the same time adhering to the same XML standards, which facilitates successful interoperation with any service consumers sticking to the same standards. In this aspect, the webservices technology is truly platform and vendor neutral as it allows each vendor to incorporate webservice support seamlessly in its platform products.

4 A Webservices Based eGovernment Framework

With the requirements in mind, we are ready to outline a webservices based eGovernment infrastructure. We will do that with a number of models:

- the system component model
- the system access model
- the security model
- the transaction model.

4.1 System Component Model

To build an eGovernment infrastructure, we need to turn the legacy government applications into service components that are to be deployed as webservices. Each of the identified webservice will be described by a WSDL document, which will be made available to the parties who have to use the services.

This exercise can be done in many different ways. Specifically, we can provide webservices to access business services, data services or objects [Linthicum 2000]. However, a reasonable approach is to extract the existing business functions provided by the legacy system. The rationale behind this is that webservices technologies are about integration. It thus only makes sense to webservice-enable a software service if there will be demands to consume it by another software system. Based on the requirements we have touched upon, it is likely that we will have the following types of webservices:

- internal services - this is the application of webservices technologies to facilitate integration of systems within a government body. For example, exposing the electronic filing system as webservices would allow automatic filing of the documents generated in the steps of processes handled by other systems of the department. This type of webservices are also useful for the construction of the department website as the exposed department services can easily be connected to the web server.

- inter-department services – this allows transactions to be conducted electronically between government bodies. For example, a central eProcurement webservice can be provided so that the system of any department can submit and track procurement orders automatically. This is useful for inter-department (G2G) process streamlining to eliminate manual paper flow between government departments, as well as the implementation of joined-up government projects which either require connection of government services to portal or delivery of macro processes involving steps spanning across different departments.
- public services – this allows easy consumption of eGovernment services by the business information systems in the private sector and is useful to support B2G applications. Moreover, an eGovernment webservice can be embedded in a business application to be provided to the citizens bundled with other value-added services provided by the business, eg the vehicle registration service can be provided by the car dealer's system to the buyer of a new car.

4.2 System Access Model

The system access model covers how the webservices will be accessed. The model addresses both the network environment [Coulouris et al 2001] of the service and the way the service is exposed to the end user.

The internal eGovernment webservices are perhaps the simplest ones in term of the access environment. The primary purpose of this webservices is to support backoffice integration. A deployment on the local area network (LAN) connecting to the other systems of the same organisation is usually sufficient for these services. The access to these services will be limited to trusted systems. As such, for these webservices, security and network performance are not major issues. The support of wireless device on the LAN does present some problems, however, but this can be circumvented by requiring all accesses from these wireless clients to go through a trusted component on the LAN.

The second type of eGovernment webservices are those provided for inter-department integration. In most countries, a government or at least state level intranet is available for deployment of these services eg the Unitary Network in Italy as described in [Mecella & Batini 2001], which should provide sufficient network performance and security protection for the applications. However, as we have discussed, access of information between government bodies is a tricky area that data privacy must be strictly observed. Hence, the protection offered by the physical security of the government intranet may not be sufficient for the more sensitive webservices.

The last type of eGovernment webservices are those that need to be available to the general public. These will be deployed over the Internet and requires strong security protection similar to the eGovernment services provided on the web. Unlike the services provided within the government networks, the limitation of the current Internet in the quality of service is an important consideration for the more critical webservices in this category.

Depending on the nature of the webservice, the consumer of the webservice may not be the same as the end user. In case both the consumer and the end user are the same, the webservice can be exposed directly to the end user. Otherwise, the webservice can only be exposed to the end user via the consumer, which would probably do that by its user interface. For example, a vehicle registration eGovernment webservice (a B2G webservice) can be exposed to the car dealer's system (the consumer) which in turn provides the service to the buyer of a new car (the end user). As we will see, webservices exposed to the end user in this indirect manner poses a special issue for end user authentication, which is not present in the traditional web applications.

4.3 Security Model

The object based access model can be used to describe the webservices infrastructure. Under the object model [Eckel 2000], we view the eGovernment webservices as interfaces to which government processes (objects) encapsulating protected information/services can be accessed. Protection of the government resources is thus achieved via the protection against unauthorised access to the webservices, according to the right access policy. We will focus on two components of the access models for the eGovernment: the access right model and the authentication model.

4.3.1 Access Right Model

Depending on the nature of the webservice, there can be more than one parties involved in using a webservice as sometimes a service is used by an end-user different with the owner of the consumer process. However, as there is always a consumer coming into play, it is easier to define the access level based on the consumer of the webservice. There are three levels of access possible:

- the consumer has no access to the webservice;
- the consumer has limited access to the webservice;
- the consumer has full access to the webservice.

For public webservices, access is effectively granted to all the consumers that can connect to the service via the internet. In addition, all these consumers will have full access to webservices offering no personal service, including the majority of information access or querying services. User level access right will be enforced, however, for the public services that involve personal information.

Similar considerations apply to the G2G webservices. There will be a group of infrastructure or administration related services that all the government department should have full access on. For the other webservices provided on the government intranet, the formulation of access policy is a more complicated exercise that should be based on the legal framework relevant to the resources encapsulated by the webservice. The requirement is that access to a webservice offered by another department should be granted if the access to the information encapsulated is explicitly approved in the law; or authorisation to access the information is available from the citizen involved. In the latter case, only limited access should be granted appropriate to the authorisation received from the citizen.

To assist the work in this area, we will need automatic tool to extract the access policy of government resources implied from the statute law into a form that can be used by the eGovernment webservices infrastructure.

For internal webservices offered within a government department, access policy can be formulated based on the organisation structure of the department, similar to other information resources deployed on the department LAN.

4.3.2 Authentication Model

The access right model is about who should do what: in the context of a webservice, we need to decide what authorities the consumer (maybe with or on behalf of an end-user) owns. Obviously this requires the identification of the consumer and sometimes the end-user to the webservice, which is the responsibility of the authentication mechanism [Oellermann 2001].

Based on the access right model, there are different cases to be handled:

- No authentication – Authentication is not required when full access of the webservice should be granted to all consumers that can connect to the service. Indeed, in this type of situation anonymous access to the webservice is the most appropriate access mode to achieve maximum privacy protection.
- Authentication on the consumer – For webservices which full access is granted based on the identity of the consumer, we need to identify the consumer only.
- Authentication on both the consumer and the end users - This is required both to support the access right policy requiring the end-user authorisation, or the use of webservice business transactions.

The requirements are straightforward except the last one. Here the challenge is to authenticate someone without a direct connection, as all the information flow between the end-user and the webservice is routed through the consumer process. We see that we have two alternatives: either we entrust the consumer in the process, or we require the end-user to use a client agent so that advance cryptography technique can be applied to supply a credential to the webservice provider.

In the first alternative, we will rely on the consumer to authenticate the end-user without mandating the mechanism the consumer should adopt. As a result, the consumer can choose the most appropriate means to decide the identity of the end-user, ranging from physical presence to the use of passwords over the Internet. This can provide the greatest flexibility to the end-user and is appropriate when there is a trusted relationship between the webservice provider and the consumer, as in the case of the G2G or G2B webservices. When the required trust cannot be established with the consumer, however, we have to rely on technologies based on digital signature which will nevertheless require a more sophisticated client agent and the establishment of a PKI in the eGovernment framework.

Another consideration for end-user authentication is on the use of single or multiple digital identities. On the system point of view, the use of a single global digital identity for each user will allow more efficient processing of the services, especially for services requiring collaboration among systems from different government bodies. However, as discussed before the citizen user may have more concern over the privacy issue and would prefer to use a separate digital identity for each different application.

4.4 Transaction Model

We have seen that the webservices technology is a useful tool to implement integrated government services, an important element of joined-up governments. There is however a transactional nature of these services that we have to address in the webservice infrastructure. Specifically, we need a mechanism to ensure that the work done by individual webservices components of the integrated service will be completed in a consistent state.

The traditional distributed transaction model is based on the atomicity of business transactions, which means that all the steps in a transaction must be completed or unwound together [Härder & Reuter 1983]. To incorporate the capability in webservices without sacrificing the interoperability we will require additional webservice standards on how coordinating information can be exchange among webservice providers. The Business Transaction Project [OASIS 2002] is a piece of work to address this issue.

Until the completion and the wide acceptance of the webservice transaction standards, we will have to adopt an alternative model based on compensating operations [Korth et al 1990]. In particular, we will need to provide a compensating webservice to each one that can take part in a webservices transaction. In addition, there should be a detection mechanism to report webservices failures, eg an end-of-day reconciliation between the requests submitted by a consumer and that processed successfully by the provider. When one of the webservices in an integrated service failed, the appropriate compensating webservice can be used to unwind the completed one. As the nature of most government services does not required the enforcement of the transaction atomicity in real-time, this should be a practical approach to apply in the eGovernment context.

We have thus outlined a webservice eGovernment framework in terms of the different webservices models. In what follows, we will focus and explore on the part of the framework that deals with the access control.

5 eGovernment WebServices Access Control

To safeguard the government resources in a webservice environment, we need to control the accesses to the government webservices. With the control in place, whenever there is a request (from outside or within the government domain) to use a government webservice, the access will be accepted or rejected according to whether the requesting party is authorised to use the service.

The access right model we have described above is to determine who should be allowed to access each webservice according to the access policy. With an access policy, we will need an access control mechanism to implement the access control according to "who should do what" as specified in the access policy.

A key issue in the webservices based eGovernment system is that we need an infrastructure to support the access control requirement. The webservices technologies are to achieve interoperability between the government systems. The implementation of the access control at the individual departments or webservices level will however likely to introduce new interoperability problems which could defeat the whole purpose of using the technologies.

To better understand the requirements of such a global access control mechanism, we will have a closer look of the following components:

- eGovernment webservice
- access to eGovernment webservice
- eGovernment access policy

5.1 eGovernment WebService

In a webservices based eGovernment system, government services are provided to the users in the form of service components deployed as webservices. As we have discussed in the system component model, different types of eGovernment webservices can be implemented to serve different purposes. These include internal services provided within a government body to facilitate system integration, inter-department services provided across government departments to support inter-department process streamlining, or delivery of citizen services requiring cross-department processing, and public services provided to allow government services to be embedded in private business systems.

In developing our security model for government webservices, we have argued that different access control requirements apply to each type (internal, inter-department, public) of government webservices. For example, a public government service that involves citizen data access would require citizen level access right, to ensure that each citizen can only access his/her own data via the webservice. The access control mechanism should thus support authentication of the webservice user identity. On the other hand, a G2G webservice not involving citizen data may be accessible by all or selected government departments, and thus the access right would solely be determined by the consumer of the webservice.

The diversity of access control requirements must be adequately addressed by the eGovernment access control mechanism. In particular, the mechanism must provide adequate flexibility to express the different types of webservice access rules precisely and efficiently. More important, the mechanism should be able to provide a level of infrastructure support such that the webservice developers do not have to worry about the access control. In other words, this should allow each webservice to be developed without any knowledge of who will and should access it. The webservice developer should focus on the business logic of the service; the mechanism to grant or reject an access request to the webservice should be the sole responsibility of the access control mechanism and must not be "hardwired" in the webservice. The application of this "separation of concerns" [Lopes & Hursch 1995] principle is to isolate and thus protect the webservice from any changes in access policy. Any changes in the access policy related to a webservice (eg a change of access right due to a reorganisation) should only affect the access control setting for the webservice and should require nothing more than the modification of its access control parameters. In what follows we will refer to the requirement to allow a webservice to be developed independent of the access policy as the "policy neutral" requirement.

5.2 Request of Access

The access control mechanism is responsible to determine whether each access request to a webservice should be accepted or rejected. In terms of the network environment, a webservice request can be made from a system connected to the same LAN as the webservice provider, a system connected to the government intranet, or just from the Internet. As one of the requirements, the access control mechanism should support access rules based on the means of access.

To illustrate the requirement, let us look at an internal webservice that provides electronic filing service to other electronic processes of the same government department. It is likely that this webservice should be provided only to systems of the department which owns the electronic filing. This would translate into an access policy that only access requests from the department LAN should be accepted.

Another example would be an eProcurement webservice offered by the central government procurement department. While this service should be offered to a much wider user base than the electronic filing webservice in the previous example, the access should still be limited to within the government. The access policy for this service will have some parts tied up with a requirement that all access requests not from the government intranet should be rejected.

The use of Virtual Private Network (VPN) technology [Scott et al 1999] to provide access to the LAN or government intranet makes the requirement a bit more complicated. As the technology can provide a secure connection over the Internet, VPN can be employed to provide low cost access to remote offices or staff at home. In effect, although the actual physical access is from the Internet, the VPN technology serve to hide this and the access should appear to be within the LAN or intranet for all security management purpose. We will take the same view and treat a VPN client/client system as part of the LAN or intranet connected to the VPN. The fact that a VPN access request comes from the Internet is irrelevant to the access control decision and should be transparent to the access control mechanism.

Association of access rights based on the means of access is a useful way to capture the eGovernment webservice access policy. However, a more general access policy should have rights associated with the parties requesting for the service. This is in close analog with the access control commonly employed on a time-sharing system, which provides access rights based on the user. To access a resource on the system eg a database file, a user should first request the necessary "privilege" of the file and subsequently all processes associated with that user will be granted access to the file.

Unlike the access to an object in a time-sharing system, an access to a webservice can be associated with more than one parties (the equivalence of the "system user" in a time-sharing system). Therefore a single party based access right such as the capability list [Lampson 1969], may not be adequate to accommodate the right access policy.

This can be illustrated with a webservice example. As a legal requirement in Hong Kong, the owner of a private car must submit his/her car for an inspection for roadworthiness every year starting from the seventh year after the manufacture date. The inspection service is outsourced to a few testing centres, who will issue a Certificate of Roadworthiness to the vehicle owner if the car passes the tests. The certificate is required for the subsequent vehicle licence renewal. Suppose that the process is streamlined to eliminate the physical Certificate of Roadworthiness. To achieve this, a webservice can be provided to the test centres to update the examination record of the vehicle directly. Renewal of the vehicle can then be based on the updated record and the certificate will no longer be needed. Such webservice will be accessed by the system of the test center as a consumer, acting on behalf of the vehicle owner as the end user.

Since the webservice would update the vehicle record, access to the webservice should be granted only if the request is made on behalf of the vehicle owner. In addition, the service should only be accessible from the test centers, not from anyone on the Internet. As such, the access should be granted based on the identity of both the end users and the consumer.

There are other possibilities. For instance, access right may be determined based on the identity of the consumer only, as in our previous example of the electronic filing webservice, which should serve all the systems on the same LAN. On the other hand, for most public citizen information enquiry webservices, eg a tax assessment enquiry service, it is sufficient to have end user based access right as the identity of the consumer is not important for the access decision.

A more complicated (and interesting) situation arises when a service is requested as a result of a chain of webservice execution. The chained webservice model is useful to support processes that require collaboration from different government departments. In this situation, when an end user initiates the process, a request will be made to access the first webservice on the chain. As all the subsequent webservices except the last one in the chain need an additional service from another department to fulfill its function, each will request for another webservice along the chain, until the end of the chain is reached which will be a webservice that can completed on its own. In term of the access right, we should view each access request as made jointly by the owner systems of the webservices on the chain preceding that webservice. If the webservice is accessible by any one or more of these systems, the access should be accepted. In other words, we should combine the access rights of all the consumer systems when there are more than one consumers involved in an access request.

Obviously, the eGovernment webservice access control mechanism must be able to accommodate each of these situations as part of its overall requirements.

5.3 eGovernment Access Policy

As a useful analogy, an access control mechanism for a webservice is like a security guard for a restricted area: while the security guard determines who should be admitted through the gate, the access control mechanism should determine which access request to the webservice should be accepted. To fulfil this responsibility, the security guard needs to know beforehand the list of people who should be admitted. Without such a list even the most experienced security guard cannot be expected to make the access control decision correctly. To complete the analogy, a similar list is required for the access control mechanism, which is the access policy in the form of the criteria for acceptance or rejection of each webservice access. The responsibility of the access control mechanism is to enforce the access policy of the webservice.

The data access policy for the systems of any modern governments is inherently a complicated subject. Firstly, the size of the problem is huge in terms of the number of potential users (including all citizen, civil servants and government departments) and the number of government systems. Also, government systems and databases are largely developed independently of one another usually with little consideration for security interoperability. As a result, it is likely that significant effort needs to taken to tackle the problems of semantic heterogeneity and conflicting security policies in the different government domains [Joshi et al 2002] to work out a government access policy on a global basis.

Perhaps not as a surprise, the adoption of a webservice eGovernment infrastructure, the primary objective of which is to solve the general interoperability issues of eGovernment projects, also helps to resolve the security interoperability problem. This is because, by extracting into webservice components only those existing business functions provided by the legacy system, a webservice based eGovernment system can be viewed as a group of business objects accessible by potential users and systems and the access to information resources are though the well-defined webservice interfaces. This eGovernment webservices component model offers a simpler conceptual model for the data access, as compared with the interconnected heterogeneous federal system model, which makes it much easier to tackle the security interoperability problems. Let us examine how the webservice access policy should look like in our webservices framework.

Property	Administration Based Policy	Legislation Based Policy
Source	Administration Decision	Legal Framework
Scope	Within Service Provider	Across the whole Government
Subject	Specific Webservice	Legal Subjects
Right For	General or Specific User	General User or Specific Government Bodies (Not Specific Individual User)
Conflicting Policies	Not Possible	Possible
Open Policy	Not In General	Yes

Table 5.3: Property of eGovernment Access Policy

5.3.1 Administration Based Policy

Based on our webservice access right model, it is useful to distinguish between two types of webservice access policies. The first type covers those access rules that are based on administrative arrangements at a department level. Most of the access rules for the internal webservices fall in this category as the access to these webservices should be formulated based on the organisation structure of the webservice owner department. Also, administration based access rules may be derived from explicit arrangements (eg in the form of a legal contract or inter-department agreement) between the webservice provider and another party to provide the business service to that party. For example, if the transport department may have an outsourcing agreement to subcontract the vehicle inspection service to a private operated testing centre, the testing centre should be granted with the access right to the appropriate vehicle record update webservice as a consumer, on behalf of the vehicle owner. Lastly, this type of policy also applies to the infrastructure or administration related webservices as the right to access the services is covered by the service level agreement (also known as performance pledge in Hong Kong) committed by the service department.

The above examples reveal a few properties of the administration based access policy. Firstly, it is to deal with the webservice specific access as the access right is based on the service to be provided to the potential consumers/users. Therefore, we will have specific access rules for each webservice. Secondly, security interoperability is not an issue here, as all the access decisions can be made within the department, either based on the organisation or explicit agreements with outsider. Lastly, there is no mandatory need to disclose the access policy in force to parties outside the department. In some cases, the disclosure is actually undesirable as it will reveal some sensitive organisational information of or confidential arrangements made by the service department. Therefore, it is more appropriate to handle the administration based access policy within the service department.

5.3.2 Legislation Based Policy

Administration based access policy is not adequate when the provider of the webservice does not have the authority to decide all the access rights. In this case, we need a source of access policy that can be applied in the global government level. This is when the legislation based access policy comes into play. It is a norm in modern governments that a comprehensive legal framework should be in place to provide the legal basis for all government activities, and in particular the interaction between government bodies and the citizen or among different government bodies. The formulation of access policy beyond the provider's domain can and should thus be based on the legislation relevant to the government service provided.

In the webservice environment, the legislation based access policy dictates access right to a webservice if the access by the user to the business service offered or the information resources encapsulated by the webservice is explicitly approved by the law. The law does not however specify the access right in the form of webservices. Instead, the security subjects (the business service/information resources) are described by legal terms in the relevant pieces of legislation. As such, legislation based access policy is based on the legal subjects and specifies who can access these subjects. The full set of legislation based access policy can thus be viewed as a set of the legal subjects and the authorised parties to access each of them. It is thus important for any access control mechanism to support an easy translation of the access rule from the policy extracted from government rule books or statutory documents.

To apply the policy by a webservice access control mechanism, a mapping between the security subjects and the legal subjects is necessary. This mapping can be a many-to-many one, which means that multiply legal rules can apply to a single webservice, and one rule can apply to more than one webservices.

The task of working out the mapping for all security subjects is a complicated task requiring significant efforts from both technical and legal experts. In our component model, however, eGovernment webservices should be based on existing business services and it is likely that the provider of each webservice is familiar with the relevant legislatures and thus the legal subjects that should apply to the underlying business service. As such, it is unnecessary to work out the mapping from the scratch. This is certainly an advantage of the webservice based eGovernment structure.

The task is less straightforward when a webservice is introduced for a new business service. In this case, the most difficult part of the task is to locate the relevant statutory applicable to the service. In reality, however, most government departments are established based on a well-defined piece of legislation. For example, the Inland Revenue Department and Company Registrar of Hong Kong are based on the Inland Revenue Ordinance (Chapter 112 of Laws of Hong Kong) and the Companies Ordinance (Chapter 32 of Laws of Hong Kong), respectively. It is highly probable that the relevant legal subjects of the services provided by a department are covered in the corresponding legislation, e.g. the submission of tax return to the Inland Revenue Department is covered in Part IX of the Inland Revenue Ordinance. Accordingly, if a webservice is to be set up for tax return submission, each citizen should have the access right to submit his/her own return.

The legal framework does not only provide an authoritative source for access policy. It also provides a common vocabulary to describe the security subjects and thus serves to resolve the problem of semantic heterogeneity in different eGovernment domains. In addition, provided that the legal framework is reasonably well integrated and self-consistent, the chance of conflicting security policies derived is minimal (Nevertheless, there is still a need to resolve the possible inconsistency within the legal framework and we will return to this point later).

The legal subject referred to in a legislation based access rule may either be specific or general. The power to obtain the tax return is an example of a rule with a specific subject. Another example can be found in the Land Registration Regulations of Hong Kong (Chapter 128A of Law of Hong Kong) which stipulates in section 4 that the "Land Registry" should be open to the public.

An example for the general legal subject can be found in Hong Kong's Personal Data (Privacy) Ordinance (Chapter 486 of Law of Hong Kong) which spells out the access right for all "personal data" as any data that "relating directly or indirectly to a living individual and from which it is practicable for the identity of the individual to be directly or indirectly ascertained and in a form in which access to or processing of the data is practicable" [HKSAR 1996]. The rule is that the "data subject" of the each piece of personal data should be allowed to access the data. As such, any webservices encapsulating personal data should be granted with access to the subject of the data. This clearly applies to a large range of government services.

A similar example can be found in the Official Secrets Ordinance of Hong Kong (Chapter 521 of Law of Hong Kong) which covers sensitive information relating to security, intelligence and criminal investigation. In effect, no access to these data from anybody should be allowed other than those working directly with the data as part of their duty.

Normally, the law provides right to access to the general public (the "Land Registry" example), the data subject of the service (the "Personal Data" example), or a specific government body. The right of a tax assessor of the Inland Revenue Department in Hong Kong to obtain all tax related information of a citizen as stipulated in section 51 of the Inland Revenue Ordinance (Chapter 112 of Law of Hong Kong) is an example of the right provided to a specific government body. Unlike the administration based policy, the legislation based policy never spells out right to individual service users. This is an important observation which we will rely on later in our proposed access control mechanism.

Another interesting point is that there can be more than one source of access policies applying to the same legal subject. This will result in conflicting access rules applied to the same webservice. This is actually the result of inconsistent legislation and there is no easy way to resolve other than the request for a court decision. Nevertheless, for legal system which provides a hierarchical legislation structure (for example, the US Constitution takes precedence over all other forms of law in the state), some priority system can be adopted and this should be catered in the access control mechanism.

It is worthwhile to point out that legislation based access policy is not something new that is invented for the webservice based eGovernment structure. The legal framework is a key part of any modern government infrastructure and there must be an element of the legislation based access policy in any eGovernment structures. However, as illustrated above, the eGovernment webservices component model, which views each webservice as a well-defined interface to the security subjects, makes it easy to utilize this type of access policy.

A summary of the two different types of eGovernment access policy is given in Table 5.3. Due to the different natures, it is more appropriate to handle each of these access policies in a different way. However, when both types of policies apply to the same webservice, we can refer to the webservice specific (administration based) rules first which is set up by the service provider. If the rules cannot determine whether an access request should be accepted or not, the subject (legislation based) rules can then be consulted.

6 Research in Access Control

This section provides a review of the existing work on access control. Each study will be briefly described and examined in the context of the applicability to the webserviced based eGovernment environment.

6.1 Traditional Model

Our study of the traditional model is based on the paper from Lampson [Lampson 1971].

The traditional access control mechanism can be modelled by an object system with three components: a set of objects *X*, a set of domains *D* and an access matrix *A*, which contains a row for every domain *d* and a column for every object *x*. Objects are the things to protect, which can include program and data files. The domains related to the process execution environments with distinct access rights. The domain concept can cover ideas like user profile and other protection contexts.

An authorization is specified by a triple $<d, x, p>$ to state that the domain *d* is authorized to exercise privilege *p* on object *x*. To set up the authorisation, the value of *p* will be stored in the access matrix *A* in the row and column corresponding to the domain/object, ie A_{dx}. Access right decision can be made by looking up of the corresponding entry is the matrix.

Some popular implementations of the access matrix will keep the authorization as a list of capability for each domain *d* which is the *d*th column of the matrix. Alternatively, an access control list may be kept for each object *x*, which is the *x*th row of the matrix. Both serve to simplify the task of maintaining and accessing the potential large but sparse access matrix.

The access matrix is a fairly low level mechanism which maps directly to the concepts in a distributed operating systems. This is a great advantage as implementation of access matrix can be easily accommodated in an operating system. The model however does not take into consideration of the business access policy and thus it has to rely on the security administrator to translate the access policy into the access matrix. This can be a formidable task, particularly in a webservice based eGovernment environment.

For example, the concept of domain, while useful in an operating system context, is too primitive to handle the access to webservices. Here we need a mechanism that provide abstraction to cater for the access right based on combination of environment factors including physical means of access, identity of the user and identify of consumer. In addition, the object concept requires all authorization to be expressed based on the operating system objects is also too restrictive to cater for the requirement of the legislation based access policy, which is based on the legal subjects instead of physical information resources.

6.2 More Advanced Models

6.2.1 Role-Based Access Control Model

Role-Based Access Control (RBAC) is a generalized access control model that provides many advantages over the traditional model. Our study is based on the paper from Ferraiolo and Kuhn [Ferraiolo & Kuhn 1992].

The concept of RBAC is based on the fact that access decisions in an organization are often based on the roles individual users take on within the organization. To apply this concept, a RBAC mechanism defines access rights to the information resources by well-defined roles. The basic scheme consists of three components: a set of well defined roles *R*, a set of subjects *S* and a set of transactions

T. The subject set *S* corresponding to the users of the protected system and is similar to the concept of domain in the traditional model. Each transaction *t* represents a transformation procedure on specific information resources. In the RBAC model, access decision is based on transactions, ie access right is expressed as whether a subject can execute a transaction. The incorporation of transaction in the model is to capture the same transaction concept in business system as a convenient base for assigning authorisation, although the model can easily be extend to cover access right based on objects.

Specifically:

- a subject *s* is authorized to perform one or more roles in the set $RA(s)$;
- a subject *s* can invoke one of the roles in $RA(s)$ as its active role $AR(s)$;
- a role *r* can be authorised to perform one or more transactions in the set $TA(r)$;

When an access decision is required to determine whether subject *s* can execute transaction *t*, the system will check wither the active role of *s*, ie $AR(s)$ is authorized to perform *t*. More formally the access should be accepted if $t \in TA(AR(s))$, and rejected if $t \notin TA(AR(s))$.

Authorisation is thus based on the roles and the access right set up involve two steps: specification of transactions for each role, and the assignment of members to the role, which are implemented by putting values in the set *TA* and *RA* respectively.

Role is a useful concept within a well-defined organisation structure when every member has clear and specific responsibility based roles assigned. As such RBAC is a powerful and flexible model to express the access policy of most business organisations.

To apply the RBAC in the government environment in a way that security interoperability can be achieved, however, would involve the set up of global roles that can be assigned across government domains. While it is possible to define some general roles (eg *GOVERNMENT and *CITIZEN as defined in our proposed model in the next section), a comprehensive role structure is not immediately available. In fact, the legislation framework which is our source for global government access policy does not seem to provide a convenient way to support this task. Unless an alternative means is available to provide the basis of the global role set up, it is difficult to apply RBAC in the eGovernment environment.

The role concept has the same limitations as the domain concept to address the need to cater for the access right based on combination of environment factors including physical means of access, identity of the user and identify of consumer.

Lastly, we observe that there is a parallel between the RBAC transaction concept and the concept of webservice. In our webservice model, each eGovernment webservice can be viewed as an object that provides a service via operations on protected information/services. Protection of the government resources via the protection against unauthorised access to the webservices is thus a similar concept to set up of authorization based on transaction. Furthermore, if we assign an implicit role to each webservice (ie one unique role for each webservice), our proposed access control model can be viewed as providing a solution to assign the membership to these roles.

6.2.2 Task-Based Authorisation Control Models

The Task Based Authorisation Control (TBAC) family of models is proposed in Thomas and Sandhu's paper [Thomas & Sandhu 1997] to address the access control requirement in workflow-based processing. The authors argue that in such a environment, permissions to access protected resources should be controlled and managed in such a way that they are turned-on only in a just-in-time fashion, according to the progress of the processing task.

In the TBAC models, authorization is granted and controlled via authorization-step. To obtain access to a protected resource, a process need to invoke the appropriate authorization step, which can be granted from a number of pre-defined trustees of the authorization step. A key feature of the models is that there is a life-time associated with each invoked authorization step, based on either usage or time parameter. TBAC also allows interrelationships to be defined between authorization steps in the form of dependencies.

There is still a lot of work required before TBAC can be applied to a real world problem like the eGovernment enviornment. For examples, a formal means is needed to specify the access policy according to the models.

In term of the webservice environment, our proposed system will capture one of the key properties of the TBAC models: just in time authorisation. This happens in application of the eGovernment webservices to implement cross-domain workflow. As access control decision in our model is made on the basis of the individual webservice, the required access right to the protected resources required for a service will only be available within the webservice. Upon the completion of a workflow step and control passed to the subsequent webservice, the access right will no longer be valid and effectively revoked.

6.2.3 Digital Library Authorisation Model

As pointed out in Bertino et al's paper [Bertino et al 2002], a Digital Library (DL) is characterised by a dynamic user population, often making accesses from remote locations, and by an extraordinary large amount of information. The traditional access matrix model is thus inadequate to handle the access control requirements in a DL system. The Digital Library Authorisation Model (DLAM) was developed by the author for the specification and enforcement of access rights applicable to the Digital Library (DL) environment.

DLAM consists of four components: subjects, objects, privileges and authorization, which roughly correspond to the concept of domains, objects, privileges and the access matrix entries in the traditional model. However, DLAM is more flexible and powerful in that it provides additional formal means to specify each of these to capture the access policy of the DL.

In case of the subjects, DLAM associate a number of DL specific roles (eg subscriber, author, co-ordinator, reviewer) and attributes to each user (eg name of the user, area of interest of a subscriber).

The content of a DL object is represented in DLAM by associating with it one or more "concepts". "Concepts" are organised hierarchically into an ontology, with the more general one at a higher level of the concept tree.

An authorisation consists of

- *subject* specification which may either be the name of the user or a credential specification which is an expression of conditions based on the roles and/or attributes associated with the subjects. As such, we can say the DLAM extend the RBAC role specification to provide a more flexible means to specify the access rules.
- *Object* specification which may either be the name of the DL object or a entity specification, which is an expression of conditions based on the "concepts" associated with the objects.
- *Privilege* granted by the authorization

- *Sign* of the authorization: whether the privilege is granted or no.

DLAM also provides the rules for authorization propagation along the concept hierarchy and exception management to resolve conflicting authorization.

Similarly to the other existing models, DLAM cannot address the need of a webservice based system to cater for the access right based on combination of environment factors including physical means of access, identity of the user and identify of consumer. In addition, DLAM is designed as a monolithic access control architecture, and thus cannot be directly applied in the eGovernment situation where both local and global level of access controls are required.

Nevertheless, there is a lot of similarities between the access control environment of a DL and the webservices based eGovernment systems and there are many DLAM ideas that are useful in an eGovernment webservice access control model. In our proposed model, we have borrowed a lot of ideas from DLAM:

- use of a subject category to support the legislation based policy, which is based on DLAM's object concept. However, we have decided not to group the legal subjects hierarchically;
- support of the sign in the access rule is based on the same concept in the DLAM authorization
- provision of resolution rules to resolve conflicting authorisation is also based on the same concept in DLAM.

6.3 Recent Works

Recently, a number of access control models have been developed to address the various requirements of distributed processing over the Internet. These include Type Enforcement, Multiple-policy Schematic Protection (MSP), Typed Access Matrix

(TAM) model and Dynamically Typed Access (DTAC) models. While these have shown potential for applicability in the eGovernment environment, they are still in the initial phases of development [Joshi et al 2002].

6.4 Limitations of the Models

As we have seen, while existing models does provide a workable mechanism to address the access control requirements of the traditional information systems, they are inadequate for the webservice based eGovernment environment.

The main problem with the models is that they are based on the simple user concept in the traditional time-sharing system, and the access control decision can be made based on a single user party associated with the access request. An access to a webservice, on the other hand, can be associated with more than one parties and a webservice access control infrastructure must therefore be able to accommodate access policies based on combination of user and consumer identity. For example, if a webservice to update the vehicle record of the transport department provided to the police department to report vehicle theft, an access to the webservice should be accepted when the consumer is from the police department and the user is the owner of the vehicle. All the existing models failed to accommodate this access control requirement.

In addition, the models we have reviewed are non-domain specific ones. As general solutions to the access control problems, these models do not cater well to some of the eGovernment specific situations:

- There are a lot of eGovernment services to provide access to citizen data. For these services, each citizen should be authorized to access his/own data only. However, citizen level access right cannot be conveniently supported in these models and must be implemented within the webservice which implies that the webservice cannot be developed in a "access policy neutral" manner.

- As we have seen, infrastructure or administration G2G webservices should be accessible by all consumer systems connected to the government intranet. Similarly, internal services provided to support integration of systems in the same government department should allow access from consumer systems via the department LAN. These access means based access right is not supported directly in the existing models and thus must be done indirectly on a consumer by consumer basis. This requires, a separate access rule for each government or internal consumer, which is a tedious and inefficient way to specify the right.
- There is no provision for legal subject based authorisation. Legislation based access policy cannot be expressed directly in the model and must be translated into object (webservice) access rights.

To provide a mechanism that work for webservice, and to provide a solution that better addresses the need for eGovernment access control, we propose a flexible access control framework which is composed of two key components: the webservice specific access control and the subject based access control.

7 Proposed Approach

In order to meet the requirements for a secured webservices based eGovernment system, we propose that a two level access control mechanism can be used. We propose a scheme based on the access model developed by Bertino et al [Bertino et al 2002]. The original model is to address digital libraries access rights and we have made enhancements to apply to the webservice and provide the two level access control.

The mechanism is flexible and powerful enough to accommodate the access right rules for all types of internal, inter-department and public eGovernment webservices as it is able to:

- accommodate the access right rules for internal, inter-department and public eGovernment webservices;
- enable the eGovernment webservices to be developed in a "policy neutral" manner;
- accommodate access right rules based on physical means of access;
- accommodate access right rules based on the combination of the end-user and consumer identity and extensible to cover the chained webservice model with multiple consumers;
- support both administration and legislation based access policy.

The proposed two level access control mechanism consists of two key components:

- the webservice specific access control and;
- the subject based access control.

We will first examine each of the two access right control components. We will then explain how these access rights can be applied to the webservices based eGovernment environment.

7.1 WebService Specific Access Control

The first level of access rights in the mechanism are those rights associated with individual eGovernment webservices. These access rights correspond to the administration based access policy which, as we have demonstrated, are all webservice specific. This is the level of access control that should be implemented within the domain of the webservice provider.

7.1.1 WebService Access Rules

The webservice specific access rights are expressed by webservice access rules. Specifically, each rule consists of the following elements:

- a *user specification* ***USER***, which specifies the end user that the access rule applies. This may be the digital identification of a specific citizen or a government staff member, *ANONYMOUS (anonymous access), *CITIZEN (any citizen with a valid identification). Note that there is a difference between the anonymous and "any citizen" right when only access to the individual data subject is granted (see below). In this case, the former specifies unrestricted rights to use the webservice, while the later implicitly specifies that the right granted is limited to using of the webservice to access the citizen data related to the end user. As such, *CITIZEN should only be applied to webservices that involved access to personal data.

- a _consumer specification_ **_CONSUMER_**, which specifies the webservice consumer that the rule applies. This may be the digital identification of a specific government department or a private organization, *ANONYMOUS (anonymous access), *GOVERNMENT (access from the government intranet), *LOCAL (access from the provider's LAN). The last two special value is to provide for specification of access right based on the physical means of access.
- a _webservice specification_ **_WS_**, which specifies the government webservice governed by the rule. It is assumed that there is a naming system in place in each eGovernment domain to provide a unique identification for each individual government webservice within the domain.
- the _sign_ of the access right, that is whether the right to access the webservice is granted "_+A_" (right to access any data subject via the webservice) "_+I_" (right to access the user's data only via the webservice); or not "-".

We can denote a webservice access rule in the form of <**_USER_**, **_CONSUMER_**, **_WS_**, "_+/-_" >. The following examples illustrate the working of the webservice access rule:

- <_*citizen_, _*anonymous_, _ws-1_, _+I_>: a request to access the webservice _ws-1_ from any consumers will be accepted. If _ws-1_ requires access to personal data, the access right is limited to the data of the requesting citizen. This is an example of a public government webservice which requires citizen level access right.
- <_*anonymous_, _*government_, _ws-2_, _+A_>: a request to access the webservice _ws-2_ from any consumers connected to the provider of _ws-2_ via the

government intranet will be accepted. This is an example of a G2G infrastructure webservice not involving citizen data.

- <*anonymous*, **local*, *ws-3*, *+A*>: a request from any user to access the webservice *ws-3* via any consumers connected to the provider of *ws-3* via the local LAN will be accepted. This is an example of an internal service provided within a government department
- <*staff-1*, **local*, *ws-4*, *+A*>: a request from *staff-1* to access the webservice *ws-4* via any consumers connected to the provider of *ws-4* via the local LAN will be accepted. If *ws-4* requires access to personal data, *staff-1* will be able to access the data of all data subject via the webservice. In this case, *staff-1* should be a staff whose day-to-day responsibility requires access to the corresponding citizen data. This is an example of a webservice specific right granted based on the status of the user.
- <**citizen*, *consumer-1*, *ws-5*, *+I*>: a request to access the webservice *ws-5* from consumer *consumer-1* will be accepted. If *ws-5* requires access to personal data, the right granted by this rule is limited to data of the requesting citizen. This is an example of a government service that is to be embedded in the system of selected party.
- <**anonymous*, *consumer-2*, *ws-6*, *->*: any request to access the webservice *ws-6* from consumer *consumer-2* will be rejected. This illustrates the use of the "-" sign to explicitly revoke access from a particular consumer.

7.1.2 Authorisation Conflict Resolution

Resolution of authorisation conflict is required when both positive ("+A" or "+I") and negative ("-") rights have been defined for the same access. Certainly we can require that all the rules specified are consistent with one another to avoid the conflict.

However, allowing different access rights to be specified at different levels of user/consumer specificity is useful when we want to provide access to a general group of user/consumer, with a few individuals excluded. To utilise this feature, we provide an authorisation resolution rule, which specifies that a specific access right always take precedence over a more general right.

The rule of precedence is expressed in Table 7.1.2. To illustrate, suppose there are two rules: <*user-1*, *consumer-1*, *ws-1*, -> and <**citizen*, *consumer-1*, *ws*, *+A*>, both apply to an access request to *ws-1* made by *user-1* via *consumer-1*. Here, the first rule is of priority 1 (specific user and consumer) while the second rule is of priority 2 (*CITIZEN and specific consumer). As such, the first rule take precedence and thus access will be rejected on the request. In effect, this combination of rules provides access of the webservice to all citizens via *consumer-1* except *user-1*. Without the authorization conflict resolution rule, this situation would require a large number of access rules to explicitly grant the right to all citizen users individually except *user-1*.

Priority	User	Consumer
1 (most specific)	A specific user	A specific consumer
2	*CITIZEN	A specific consumer
3	*ANONYMOUS	A specific consumer
4	A specific user	*GOVERNMENT/*LOCAL
5	*CITIZEN	*GOVERNMENT/*LOCAL
6	*ANONYMOUS	*GOVERNMENT/*LOCAL
7	A specific user	*ANONYMOUS
8	*CITIZEN	*ANONYMOUS
9 (least specific)	*ANONYMOUS	*ANONYMOUS

Table 7.1.2: Priority of access rules based on the specificity of the user/consumer

7.2 Subject Based Access Control

The second level of access rights in the mechanism are those rights associated with webservice subjects. These access rights correspond to the legislation based access policy, which is based on the legal subjects and specifies who can access these subjects. Subject based access control is to be operated over the individual government departments and this is the level of access control that should be implemented at a global level.

7.2.1 Subject Category

The subject category is a repository of all the legal subjects which are the objects of access rights. The category should contain an entry for each legal subject with both the proper legal term and its source, which specifies the piece of legislature which is the source of the legal term. The source is included as reference information to the legal subject and usually it should be a pointer to the relevant interpretation section in the corresponding legislature. Inclusion of this reference provides a means to retrieve the legal interpretation of the subject when there are questions to the details of the underlying access policy later.

The subject category should provide a unique identification for each subject. The subject ID will be used in the subject access rules to associate the access rights to the subject.

The subject category is similar to the object concept structure in the model of Bertino et al [Bertino et al 2002] and provides the same functionality in our model. The major difference in our model is that we have adopted a simple set structure for the legal subjects instead of a hierarchy structure. While a hierarchy structure may provide a more powerful mechanism to express the access right, we believe that more research in the area of the legal subjects is required to conclude whether this is appropriate and feasible.

7.2.2 Subject Access Rules

The subject based access rights are expressed by subject access rules. Specifically, each rule consists of the following elements:

- a *user specification* ***USER***, which specifies the end user that the access rule applies. The use and interpretation of this element is the same as in the webservice access rule except that only two values are allowed: *ANONYMOUS (anonymous access) or *CITIZEN (any citizen with a valid identification).
- a *consumer specification* ***CONSUMER***, which specifies the webservice consumer that the rule applies. Same as in the webservice access rule.

- a *subject specification* ***SUBJECT***, which specifies the legal subject webservice governed by the rule. This should contain the unique ID of legal subject that can link up with the subject category.
- a *privilege specification* ***PRIVILEGE***, which specifies privilege granted or revoked by this rule. This may be *READ (read privilege) or *UPDATE (update privilege, which implies the *READ privilege).
- a *source specification* ***SOURCE***, which specifies the legal source of the rule. This should contain a reference to the relevant piece of legislature which is the basis of the access right.
- a *priority specification* ***PRORITY***, which optionally provides a priority number of the rules which can be used to resolve conflicting rules. This applies only to legal system which provides a hierarchical legislation structure (for example, the US Constitution takes precedence over all other forms of law in the state).
- the *sign* of the access right, that is whether the right to access the webservice is granted "+*A*" (right to access any data subject via the webservice) "+*I*" (right to access the user's data only via the webservice); or not "-".

We can denote a subject access rule in the form of <***USER***, ***CONSUMER***, ***SUBJECT***, ***PRIVILEGE***, ***SOURCE***, ***PRIORITY***, "+/-" >. The following examples (which are based on the law of Hong Kong) illustrate the working of the subject access rule:

- <**anonymous*, **anonymous*, "*Land Registry*", **read*, "*Land Registration Regulations s4*", *1*, *+A*>: specifies that any user can request via any consumer systems to webservices requiring read-only access to the Land Registry information.;

- *<*citizen, *anonymous, "personal data", *update, "Personal Data (Privacy) Ordinance s22", 1, +I>* specifies that any user can request via any consumer systems to webservices that will update his/her own personal data.;
- *<*anonymous, "Inland Revenue Department", "tax related information data", *read, "Inland Revenue Ordinance s51", 1, +A>* specifies that request will be accepted via the Inland Revenue Department's system to webservices requiring read-only access to the "tax related information data".

7.2.3 WebService Registration

To apply the subject based access right to a webservice, we need to associate the webservice with the subject(s) representing the security subjects (the business service/information resources) it encapsulates. This can be done by registering the webservice in the subject category, effectively mapping the webservice to the applicable subject(s). Note that the registration can be done locally within the provider's domain and a global registration database is not required.

A registration entry for a webservice should contain the following:

- the webservice to register;
- the subject id of the subject that the webservice should be associated with;
- the privilege of the subject that is required to access the webservice, this can be *READ (read privilege) or *UPDATE (update privilege).

Once the registration is done, the subject based access control mechanism operates by referring to the access rules of all the subject(s) associated with the webservice. A request to access the webservice should be accepted if the necessary privilege on the associated subject has been granted by one of the rules in effect.

It is possible that a webservice is registered under more than one subjects. This would happen when the security subjects are governed by more than one pieces of legislation, or when the webservice encapsulates more than one security subjects. In this case, the access test must be performed for all the associated subjects and a request should be accepted if the necessary privileges required for all the subjects are granted.

7.2.4 Authorisation Conflict Resolution

Similar to the webservice access rules, it is possible that both positive ("+*A*" or "+*I*") and negative ("-") rights are defined for the same access in the subject based access control. This may reflect the need to define a general access rule supplemented by a few exceptions, as we have seen in the webservice access control example. However, this may also be resulted from more than one sources of access policy in force for the same legal subject. In each case, we need a similar schema to resolve the conflicts in authorisation.

When there is a conflict between two rules applying to the same subject, we will first resolve the conflict by the priority of the rule. The rule with the higher priority will take precedence. If both rules are of the same priority (or no priority is specified), the "specific over general" rule can be applied, although some modification on the rule of precedence is required to take into account the different values supported by the subject based access rules. The rules are expressed in Table 7.2.4:

Priority	User	Consumer
1 (most specific)	*CITIZEN	A specific consumer
2	*ANONYMOUS	A specific consumer

3	*CITIZEN	*GOVERNMENT/*LOCAL
4	*ANONYMOUS	*GOVERNMENT/*LOCAL
5	*CITIZEN	*ANONYMOUS
6 (least specific)	*ANONYMOUS	*ANONYMOUS

Table 7.2.4: Priority of access rules for subject based rule

If all the above fail to resolve the conflict, the mechanism should reject the access and report the case as an exception (this is only possible when there is a conflict between different legislation provisions). This is the safe approach as the user can always request the business service with an alternative means.

7.3 The WebServices

The proposed mechanism does not require a government webservice to be aware of the access control policy that will be applied. The infrastructure service provided by the mechanism will decide whether a particular request to access the webservice should be accepted or not. Accepted requests will be routed to the webservice, while unauthorized requests will be rejected right away.

7.4 Combining Two Level Access Control

In general the two levels of access control works independently. The webservice access rules work in a local context within the webservice provider's domain. The subject access rules work across the government and are controlled centrally. The webservice provider's role is restricted to the registration of the webservice.

It is possible that both types of access rights apply to the same webservice. Our rule is that the webservice access rules should always be consulted first. If the rules cannot determine whether an access request should be accepted or not, the subject rules can then be consulted. This is based on the assumption that the webservice provider should take into account of all the relevant facts when the administration based (webservice specific) access policy is worked out and thus the rule should take precedence over the more general legislation based rights.

An exception will occur if both level of access control cannot determine whether a request should be accepted. In this case, the request should be rejected. This is actually an arbitrary decision based on the fact that the user can always request the business service with an alternative means.

7.5 Application to Chained WebService Request

Both the webservice and subject access rules only accommodate one consumer party. As a result, the rules cannot be applied directly in the more complicated situation when a service is requested as a result of a chain of webservice executions. We have argued that we should view such access request as made jointly by the owner systems of the webservices on the chain preceding the webservice. For the purpose of access control, we can view these requests as made with multiple consumers and we should combine the access rights of all the consumer systems. Along these lines, it is possible to extend the proposed mechanism to cover the chained webservice requests.

For the webservice access rights, we can first determine whether the request of a chained request should be accepted based on the identity of each consumer. To combine the access rights, the request should be accepted if any consumer has got the right to access the service.

Similar arrangement can be applied to the subject based rights. Here the schema should however work on the subject level. We should determine whether the required privilege on a subject is granted based on the identity of each consumer. If the privilege has been granted to any of the consumers in the request, it should be treated as granted. As such, when a webservice requires privileges to several security subjects, it is possible to access the webservice with a request made jointly by a group of consumers if their access rights add up to the required privileges, even though individually neither of them can access the webservice.

7.6 Comparison with the Existing Access Control Models

As we have mentioned in section 6.4, the existing access control models cannot address the requirement of eGovernment webservice as they fail to support access rules based on combination of user and consumer identity.

In the proposed system, both a user specification and a consumer specification are in place in the access rules. This allows specification of rules based on combination of user and consumer identity and makes the system adequate to address the webservice requirements.

In contrast to the existing models, the propose system also caters better to the eGovernment access control infrastructure with support for the following domain-specific requirements:

- citizen level access right can be specified directly with the sign of "+I";
- special consumer specification values *LOCAL and *GOVERNMENT are provided to express access rules based on physical means of access;
- the subject based access control component is in place to support legislation based access policy.

A more comprehensive evaluation of the proposed model is provided in section 9. In particular, application scenarios which illustrate the detailed working of the above in the proposed system can be found in section 9.1.

8 An Implementation Reference Model

8.1 Some Practical Issues

Before we go into details of our implementation model, it is worthwhile to bring out a few practical issues related to the implementation of a secured webservice infrastructure. They are discussed in the following subsections:

- Citizen privacy
- Trust between eGovernment systems
- Authentication

8.1.1 Citizen Privacy

We have argued that protection of citizen's personal information privacy is one of the eGovernment security requirements. Most legal systems have relevant legislation eg the Personal Data (Privacy) Ordinance of Hong Kong, to implement the protection of citizen data privacy which should form part of the legislation based access policy to govern the necessary access control to achieve the required protection.

We have also mentioned that the request for a service made by a citizen should only be made known to the service provider and not to other parties, unless the disclosure of this information is explicitly allowed by the citizen. This is to safeguard the citizen from undesirable data surveillance based on the access pattern collected without his/her consent.

In terms of the eGovernment webservice access control, this places a more subtle but less obvious requirements: the mechanism should not require the disclosure of the citizen service requestor's identity at a global level, beyond the service provider's domain. For example, a global access control engine that requires the submission of the citizen's identity to determine when an access should be granted is unacceptable. This is because that in such case the global engine will be able to collect the pattern of all accesses made by the citizen even different electronic identities are provided to the citizen, and is thus a clear violation of the above privacy requirement. This in effect requires a strong separation of the authentication mechanism with the access control decision and is an important constraint to the mechanism that can be applied. We will see how the two level access control based access control mechanism can be employed to meet this need.

8.1.2 Trust between eGovernment Systems

It is a general design principle of a security system that the reliance of trust between different systems should be minimized. The rationale behind is that any compromise in one system will affect the integrity of all systems that trust it. The avoidance of all kinds of trust in a system may not be practical. As demonstrated by Yahalom et al [Yahalom et al 1993], in reality most of the security algorithms (including Keroberos, SPX and SELANE) employ some degree of trust.

We argue that the same should apply to the task of end-user authentication when the access right policy of the webservice requires a decision based on the identity of the end-user, who makes the request via a consumer running on another government department. In many cases, the service provider is willing to trust the consumer department if it is relied on to deliver the service to the citizen. As such, the presentation of the citizen's digital identity by the consumer process is sufficient to support the fact that the request is made on behalf of that citizen. This will enable the consumer department's staff to choose the most appropriate means to decide the identity of the citizen, ranging from physical presence to the use of a department maintained user password and we have argued that this trust arrangement provides the greatest flexibility to the end-user.

The trust arrangement between government bodies is not part of the access policy: it only deals with how the access rule can be implemented. However, it is desirable for the access control mechanism to accommodate the arrangements to enable easy configuration of webservices. We suggest that the trust arrangements can be handled as part of the administration based access policy. This can be done by granting webservice specific access right to the trusted government department(s) just based on the department's identity. In effect, the authentication of the citizen is not required for the trusted department to gain access to the webservice.

If the webservice is access by another department not in the trusted list, it will not be covered by the relaxed webservice specific access right and an authenticated citizen's identity will be required based on the prescribed access policy.

8.1.3 Authentication

The access control is to determine who can access a webservice and this requires the identification of the consumer and/or the end-user requesting the service. Although the authentication model is separate with the access right model, the authentication mechanism is an important part of the security model and some discussion of it is necessary to complete the whole picture of access control mechanism.

As we have mentioned, one of the challenges for security implementation in eGovernment systems is the need to provide the citizen with choices over how much of the data he or she is willing to provide, in exchange for a more powerful and efficient service. This implies that the authentication mechanism must be feasible to allow different kind of digital identities.

For example, to provide a single signon (SSO) functionality to different webservices across the eGovernment, the authentication system must support a global identity and a common authentication entity (the entity may consist of multiple distributed servers but each must be linked to share a common identity database). Users opted to enjoy the SSO functionality will have their service request logged by the common authentication mechanism. As such, the more privacy concerned citizen may choose to have a separate identity with each government department (service provider) to avoid going through the global authentication process. In this way, the citizen can be more certain that his/her service request will not be known to other parties outside to the service provider. The use of digital identities from different government domains is thus a key requirement.

As different government departments are likely to have their legacy identity systems, the requirement for multiple identity domains necessitates the support of multiple protocols (eg Kerberos tickets, PIN, eCertificate based electronic signature, etc), according to the identity domain adopted by the citizen in the webservice access.

We have discussed the issue of trust between the consumer and the service provider in a webservice request made on behalf of a citizen. Trust between the client and the consumer is however another issue that we need to examine in terms of the authentication process. Authentication can be straightforward if there is a trusted relationship between the citizen and the consumer (for example, when the consumer is a government department) as the citizen can supply the relevant PIN or private key to produce the necessary credential to the provider through the consumer. However, if such relationship cannot be established, as in the case when the consumer is from the private sector, a more supplicated arrangement will be necessary. One of the possibility is to use a signed codelet (which is a feature supported by most common internet browsers) from a trusted party (eg the government) which can be downloaded on request to produce the credential from the citizen's password or private key. This scheme works because the citizen can trust that the information supplied to the codelet will not be disclosed to the consumer.

Lastly, to support the access control requirements we just gone through, it is not sufficient for the authentication system just to provide the identity of the end-user. To support physical access based policy, the mechanism must also be able to provide the physical access information of the request, ie whether the request is coming from the provider's LAN, the government intranet, or the internet, as well as anonymous access which should be accepted by the access policy of the public information enquiry webservices.

8.2 System Architecture

To implement the two level access control model, a typical system architecture consists of the following:

- eGovernment webservices gateway
- authentication engine
- access control database
- access control decision engine.

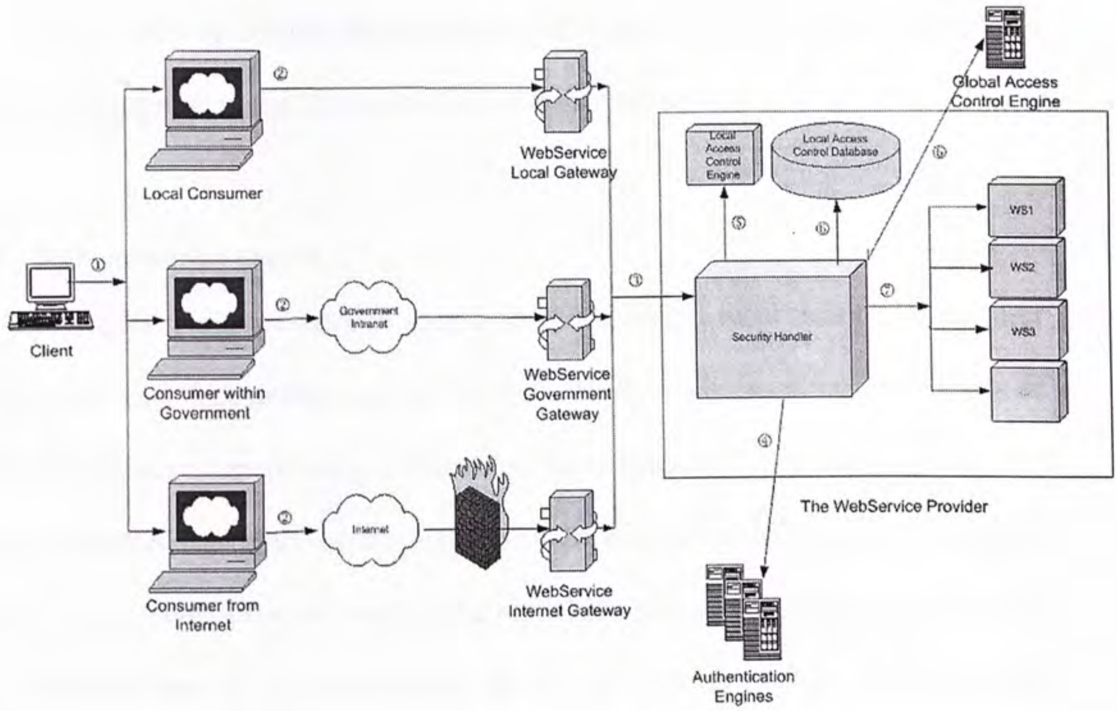


Figure 8.2 : Proposed System Architecture

8.2.1 eGovernment WebServices Gateway

To protect the eGovernment webservice resources, access to the webservices should be restricted to well-defined access points. This can be achieved by eGovernment webservices gateways which provide connection to the webservice from different network environments. A different gateway can be set up for access via the local LAN, the government intranet and the Internet.

For public service, consumer systems will able to submit the webservice request to the appropriate Internet webservice gateway via the Internet. Similarly, government consumer systems can access the eGovernment webservices via the government intranet gateway. For internal service, consumer systems connected to the webservice provider on the same LAN will request the webservice via the LAN gateway.

In each gateway, additional protection hardware and software (eg firework) can be added appropriate to the corresponding network environment.

8.2.2 Authentication Engine

The request for a webservice may contain the digital identification of the user and/or consumer. The access control decision will be made based on the identity of the user/consumer corresponding to the digital identification.

It is the responsibility of the authentication engines to provide and prove the identity of the user/consumer based on the digital identification provided. As we have seen, there is a need to support multiple identity domains, both global and local ones managed by different departments. We have assumed that an authorization engine will be available for each identity domain. The security infrastructure of provider should be able to locate and access the proper authorization engine corresponding to the digital identifications.

8.2.3 Access Control Database

To implement the access control models, a number of databases should be maintained.

In the local level, ie specific to the service provider, a database storing the webservice access rules for all the local webservices is required. This will be consulted to determine locally whether a request should be accepted based on the webservice access rights.

In addition, the local database will also include the registration information of the local webservices. To invoke the subject based access control decision, this database provides information of the subjects associated with the webservice in question, which will be required to retrieve the appropriate subject access rules.

At the global level, a database storing the subject access rules will be maintained.

8.2.4 Access Control Decision Engine

To handle the two levels of access control independently, two access control decision engines will be required. The local access control engine will make decision based on the webservice access rules. A subject access control engine which runs outside the provider's domain will be responsible to make decision based on the subject rules.

In both cases, the engine should decide not only whether a webservice request should be accepted. It also has to determine whether the access right granted is based on a data subject specific access rule (eg a rule with the "+*I*" sign). If this is the case, the webservice request should be examined to check that the data subject accessed is that of the requesting citizen.

An important feature of the global engine is that the query should be based on generic citizen identity (*CITIZEN or *ANONYMOUS) and the legal subject. The use of generic citizen identity ensure that access request information of a citizen will not be disclosed outside the service provider's domain. This is possible because the subject access rules only refer to generic user identify. Also, use of legal subjects in the access control query instead of the webservice provides another privacy protection as the global engine will not have any knowledge of the specific service the citizen is requesting.

8.2.5 A Working Scenario

To illustrate how the different architecture components would work together, a working scenario to make the access control decision for a webservice request is provided. The step number referred in the example corresponds to the same number in Figure 8.2.

1. The user invokes a web-based application on the consumer's system. The application requires the service of an eGovernment webservice.
2. The consumer system prepares the necessary webservice request, with the digital identities as required by the webservice. The request is submitted to the appropriate gateway, depending on the type of consumer.
3. The gateway routes the webservice requests to the appropriate provider, which is intercepted by the security handler of the provider's webservice engine.
4. The security handler locates and invokes the appropriate authentication engine to use based on the digital identity. The request will be rejected if authentication fails. This step may be skipped if the request is an anonymous one.
5. The local access control decision engine will be consulted to see whether the request should be accepted. The engine will make use of the local access control database to retrieve the rules applicable to the requested webservice. If the access is based on an access rule with specific data subject (ie a "+*I*" rule), the webservice request will be examined to check that the data subject accessed is that of the requesting citizen. If a decision can be made, the request will either be routed to the appropriate webservice (step 7) or rejected right away.

6. If a decision cannot be made within the provider's domain, the subject(s) associated with the requested webservice will be retrieved from the local access control database and the global access control decision engine will be consulted to make a decision. If the access is based on an access rule with specific data subject (ie a "+*I*" rule), the webservice request will be examined to check that the data subject accessed is that of the requesting citizen. The request will be rejected according to the result of the decision.
7. The request will be passed to the corresponding webservice.

8.3 Implementation

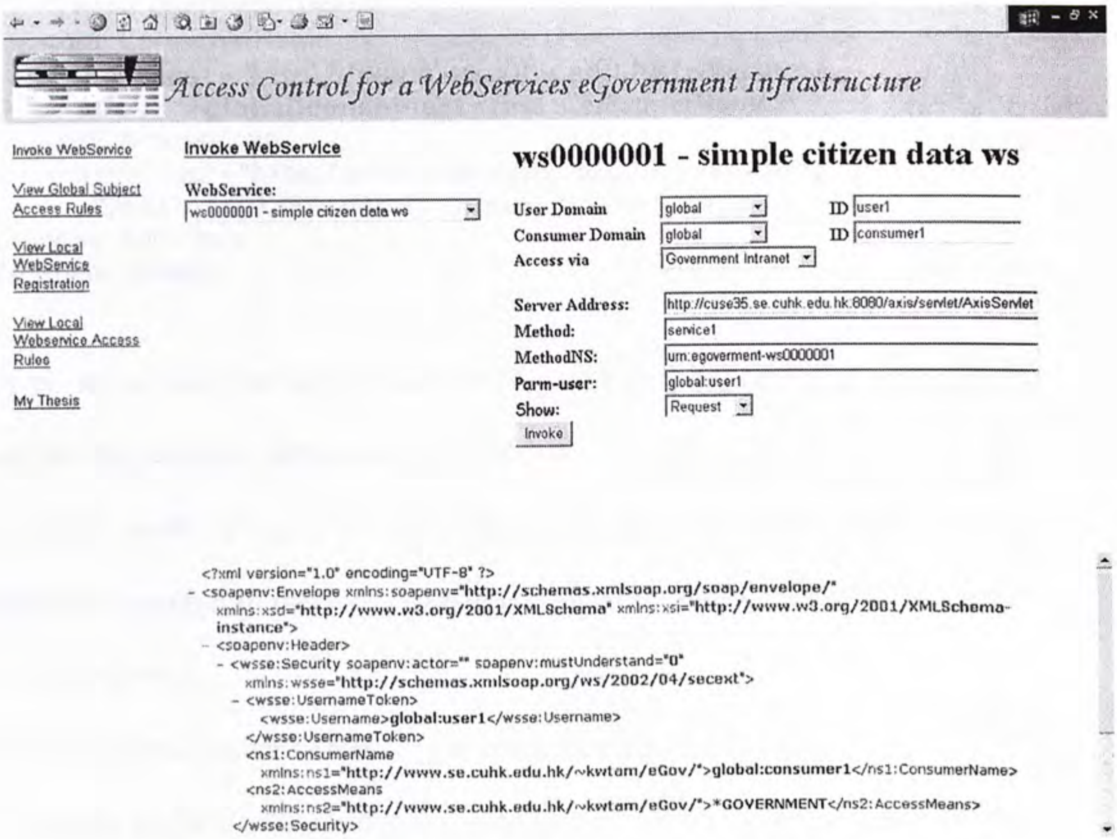
We have implemented a prototype system on top of Apache Axis using the Java platform. The developed system is largely according to the architecture described in Section 8.2, with the following simplifications:

- To reduce the demand on hardware, the three WebService Gateways (local, government and internet) are simulated instead of implemented physically. This is achieved by setting of the access medium (ie LAN, government intranet or Internet) by our webservice client based on a user input parameter.
- The authentication engines have not been implemented. This is to simplify the implementation as our focus is mainly on the access control.
- The extension to handle chained webservice requests as described in Section 7.5 is not implemented.

Figure 8.3A shows a screenshot of the system when a request is prepared for a webservice invocation. A webservice request is prepared based on the following input:

- user domain and id

- consumer domain and id
- access medium (which simulates the use of different webservice gateway, as mentioned above)
- webservice parameters


Access Control for a WebServices eGovernment Infrastructure
Invoke WebService
View Global Subject Access Rules
View Local WebService Registration
View Local Webservice Access Rules
My Thesis
Invoke WebService
Invoke

WebService: ws0000001 - simple citizen data ws

ws0000001 - simple citizen data ws

User Domain: global ID: user1
Consumer Domain: global ID: consumer1
Access via: Government Intranet

Server Address: http://cuse35.se.cuhk.edu.hk:8080/axis/servlet/AxisServlet
Method: service1
MethodNS: urn:egoverment-ws0000001
Parm-user: global:user1
Show: Request

```
  <?xml version="1.0" encoding="UTF-8" ?>
- <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance">
  - <soapenv:Header>
    - <wsse:Security soapenv:actor="" soapenv:mustUnderstand="0"
        xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/04/secext">
      - <wsse:UsernameToken>
          <wsse:Username>global:user1</wsse:Username>
        </wsse:UsernameToken>
        <ns1:ConsumerName
          xmlns:ns1="http://www.se.cuhk.edu.hk/~kwtam/eGov/">global:consumer1</ns1:ConsumerName>
        <ns2:AccessMeans
          xmlns:ns2="http://www.se.cuhk.edu.hk/~kwtam/eGov/">*GOVERNMENT</ns2:AccessMeans>
      </wsse:Security>
```

Figure 8.3A: A WebService Request

The use of the fields "user domain" and "consumer domain" is to support digital identities from different government domains as it provides the information for the system to choose the appropriate authentication engine to validate the digital identification.

Once all the information has been input into the system, a SOAP request message will be generated and sent to the webservice provider. The access control information (user, consumer and medium type) is incorporated in a header element compatible with the WS-Security [Atkinson et at 2002] standard. The following is an example of a header generated:

```
<soapenv:Header>
   <wsse:Security soapenv:actor="" soapenv:mustUnderstand="0"
       xmlns:wsse="http://schemas.xmlsoap.org/ws/2002/0
       4/secext">
     <wsse:UsernameToken>
         <wsse:Username>global:user1</wsse:Username>
       </wsse:UsernameToken>
       <ns1:ConsumerName
         xmlns:ns1="http://www.se.cuhk.edu.hk/~kwtam/
         eGov/">global:consumer1</ns1:ConsumerName>
       <ns2:AccessMeans
         xmlns:ns2="http://www.se.cuhk.edu.hk/~kwtam/
         eGov/">*GOVERNMENT</ns2:AccessMeans>
     </wsse:Security>
   </soapenv:Header>
```

On the server side, the security handler (Figure 8.2) is implemented via a SOAP Message Handler which is defined by the JAX-RPC standard. With Apache Axis, the access control can be "plugged-in" to the eGovernment webservice by incorporating the handler information in the Web Service Deployment Descriptor (WSDD) for the published webservice.

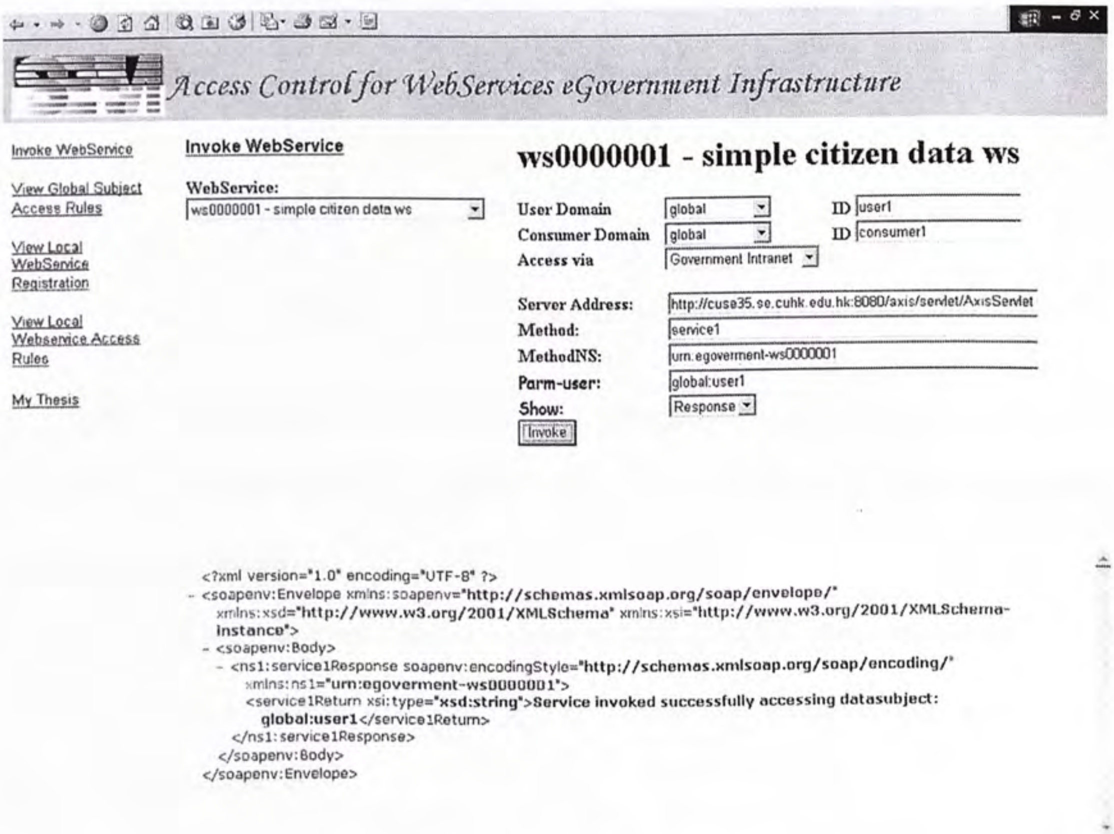
The implemented security handler performs the following functions:

- intercept the SOAP request message before the request is processed by the provider
- extract the access control information from the WS-Security header
- check the local webservice based access control database for access right
- consult the Global Access Control Engine (Figure 8.2) for subject based access right, if necessary
- check that the data subject accessed is that of the user, in case the access right is "+*I*" (this is done by inspecting the appropriate element of the

SOAP body for the request parameter that representing the data subject based to the provider)

- reject the request if the access right is not granted, or a data subject other than the requesting user is tried to be accessed without a "+*A*" right
- pass the request to the provider if the request is authorized

Figure 8.3B shows a screenshot when a request with proper authorization has been processed. The screen shows the response SOAP message returned by the webservice provider.


Access Control for WebServices eGovernment Infrastructure
Invoke WebService
View Global Subject Access Rules
View Local WebService Registration
View Local Webservice Access Rules
My Thesis
Invoke WebService
Invoke

WebService: ws0000001 - simple citizen data ws

ws0000001 - simple citizen data ws

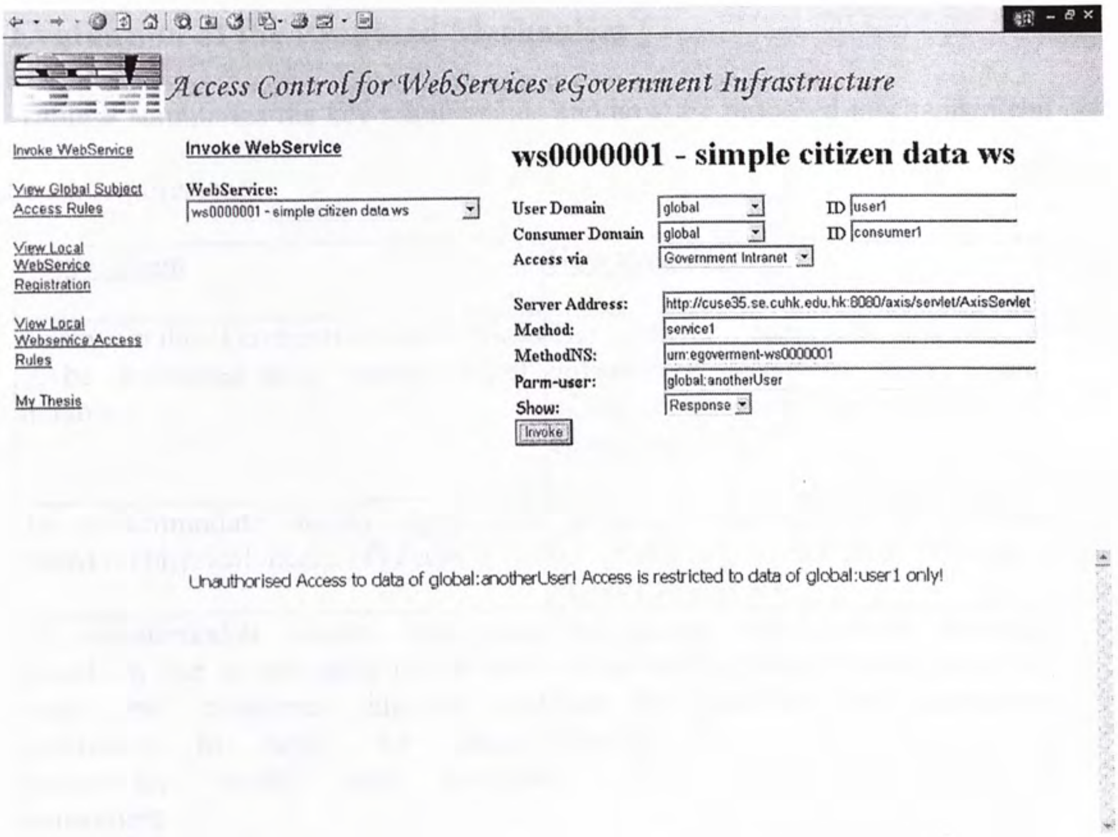
User Domain: global ID: user1
Consumer Domain: global ID: consumer1
Access via: Government Intranet

Server Address: http://cuse35.se.cuhk.edu.hk:8080/axis/servlet/AxisServlet
Method: service1
MethodNS: urn:egoverment-ws0000001
Parm-user: global:user1
Show: Response

```
<?xml version="1.0" encoding="UTF-8" ?>
- <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
    instance">
  - <soapenv:Body>
    - <ns1:service1Response soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
        xmlns:ns1="urn:egoverment-ws0000001">
        <service1Return xsi:type="xsd:string">Service invoked successfully accessing datasubject:
          global:user1</service1Return>
      </ns1:service1Response>
    </soapenv:Body>
  </soapenv:Envelope>
```

Figure 8.3B: A Successful Request

The rejection message from the security handler will be shown if the request is not authorized, as illustrated in Figure 8.3C.


Access Control for WebServices eGovernment Infrastructure
Invoke WebService
View Global Subject Access Rules
View Local WebService Registration
View Local Webservice Access Rules
My Thesis
Invoke WebService
Invoke

ws0000001 - simple citizen data ws

WebService: ws0000001 - simple citizen data ws

User Domain: global — ID: user1
Consumer Domain: global — ID: consumer1
Access via: Government Intranet

Server Address: http://cuse35.se.cuhk.edu.hk:8080/axis/servlet/AxisServlet
Method: service1
MethodNS: urn:egoverment-ws0000001
Parm-user: global:anotherUser
Show: Response

Unauthorised Access to data of global:anotherUser! Access is restricted to data of global:user1 only!

Figure 8.3C: An Unauthorised Request

The Global Access Control Engine is implemented as a separate Java function, which can either be invoked locally as a Java method, or consumed as a published Apache Axis webservice.

Both the local and and global access control databases are maintained as xml files, which are loaded by the local security handler and Global Engine respectively during each invocation.

9 Evaluation of the Proposed Mechanism

Table 9 summaries the key requirements and how the proposed mechanism can meet each requirement:

Requirement	How to achieve
To enable the eGovernment webservices to be developed in a "policy neutral" manner	The scheme does not require a government webservice to be aware of the access control policy that will be applied.
To accommodate access right rules based on physical means of access.	The access rules support rights based on means of access like *LOCAL, *GOVERNMENT.
To accommodate access right rules based on the combination of the end-user and consumer identity and extensible to cover the chained webservice model with multiple consumers	The àccess rules contain specific components to express rights based on both the end user and consumer identify.
To support both administration and legislation based access policy	The administration and legislation based policy are accommodated in the webservice and subject based access control respectively.

Table 9: Qualitative Evaluation of the Two Level Access Control Model

Based on our research, no existing access models we are aware can meet all the above requirements. In addition, an implementation based on our reference model will be able to meet the following citizen privacy requirements:

- the access request information of a citizen will not be disclosed outside the service provider's domain
- a citizen can use different types of digital identities to request for eGovernment webservice, ranging from a global identity across all government departments to those provided by individual departments on a local level.

9.1 Application Scenarios

The following provides examples of typical access control scenarios in the government environment and how the proposed mechanism can address them. This is to illustrate how our proposed mechanism can meet the various eGovernment requirements.

9.1.1 Citizen Level Access Right

Probable the most common type of eGovernment information services are those providing enquiry and/or update of a citizen's data record kept by the government department. Some of the examples are:

- enquiry of the citizen's driving license record
- update/correction of the citizen's address information
- submission of information (eg tax return)

The access control requirement for these webservices is simple: each citizen should be authorized to access his/her own data only.

The proposed mechanism offers a handy way to support this citizen level access need, as the sign field of the access rule supports value of "_+I_", which specifies that the access to the webservice is limited to the data of the requesting user.

In comparison, none of the access control mechanisms we have reviewed supports this type of access right. The user level access right can thus only be implemented within the services, which implies that the webservices cannot be developed in a "policy neutral" manner.

The following are examples of how the required access right can be defined using the webservice based access rules:

- _<*citizen, *government, ws, +I>_ provides access of the webservice to all user who can provide a valid digital id, via the government intranet

- *<*citizen, *anonymous, ws, +I>* provides access of the webservice to all user who can provide a valid digital id, via any consumer.

The use of subject based access rules to implement user level access right is similar and thus not repeated.

9.1.2 Access Means Based Authorisation

Another common type of services are those infrastructure or administration related G2G webservices, which provided infrastructure service, or support inter-department system integration. Some of the examples are:

- timestamp service provided by the central IT centre
- eProcurement service provided by the government central procurement department

This type of services should be accessible by all the parties who are authorized to access the government systems, or have access to the government intranet.

There is another type of internal services which are provided to support integration of systems in the same government department. Examples are:

- submission of document to the department's electronic filing system
- enquiry and update services to the department's operating system

Similar to the G2G services, the access to these services should be based on the access means: access should be allowed via the department's local LAN.

The proposed mechanism supports access right based on the physical means of access by the special value *LOCAL and *GOVERNMENT in the consumer specification. An access rule with a consumer specification of *LOCAL or *GOVERNMENT will provide access right to webservice requests coming from the department LAN or government intranet respectively.

While access means based access right can be supported in other access control mechanisms, it requires specification of the right on a consumer by consumer basis to do so. Our proposed mechanism thus provides a more compact and efficient way to define the right.

The following are examples of how the access means based right can be defined using the webservice based access rules:

- <*anonymous, *government, ws, +A>* provides access of the webservice to all users on the government intranet
- <*anonymous, *local, ws, +A>* provides access of the webservice to all user on the department LAN.

The use of subject based access rules to implement access means based right is similar and thus not repeated.

9.1.3 Access Right Based on Combination of User and Consumer Identity

A more general type of webservice access right will require the specification of both the user and consumer identification. Some common examples of this type of webservices are:

- G2G citizen webservice provided to a specific government department to support operation streamlining (eg a webservice to update the vehicle record of the transport department provided to the police department to report vehicle theft);
- citizen webservice provided to support operations which has been outsourced to a third-party (eg, a webservice to update the vehicle examination record of the transport department should be provided to the third-party test center which is sub-contracted to perform the vehicle fitness certification for the vehicle owners);

As both the user and consumer specification are present in an access rule, the proposed mechanism supports the access right based on combination of user and consumer identity in a flexibile and simple manner. For example, an access rule of the form <*citizen, consumerid, ws, +I> can be specified to provide access of the webservice to all user who can provide a valid digital identification, via the specified consumer's system. The access is limited to the user's data.

On the other hand, all other access control mechanisms only allow the specification of a single user party in the access rule and thus cannot easily accommodate these type of webservices.

9.1.4 Legislation Based Access Right

As we have discussed, access right to the government service should have backup from the relevant statutory laws. This can be expressed conveniently in the proposed model as legal subject based access rule. The following examples (which are based on the law of Hong Kong) illustrate the working of the subject access rule:

- <**anonymous*, **anonymous*, **read*, "*Land Registry*", "*Land Registration Regulations s4*", *1*, *+A*> specifies the duty of opening of Land Registry to the public;
- <**citizen*, **anonymous*, "*personal data*", **update*, "*Personal Data (Privacy) Ordinance s22*", *1*, *+I*> specifies the right of a data subject to correct his/her personal data;
- <**citizen*, **anonymous*, "*tax return*", **update*, "*Inland Revenue Ordinance s51*", *1*, *+I*> specifies the right of a citizen to update his/her tax return;

- <*anonymous*, "*Inland Revenue Department*", "*tax related information data*", **read*, "*Inland Revenue Ordinance s51*", *1*, *+A*> specifies the power of tax assessor to obtain information related to tax.

The legislation based access right is not directly supported in other access control mechanisms. The only way to incorporate the rights is to translate them to explicit webservice rights, which can be a tedious and error-proning process and this approach also suffers from the subsequent maintenance problem.

9.1.5 Joined-up Government

Implementation of the joined-up government requires integration of services provided by different government departments. As an example, the application of a license for a restaurant in Hong Kong requires the services from a number of government departments including the Buildings Department, Food and Environmental Hygiene Department and Fire Services Department. To provide a webservice for restaurant licensing will thus integrate the services from these departments. The access control of the integrated services should represent an aggregation of the component services.

As the proposed mechanism supports legislation based subject access rules, the access rights of an integrated service can be specified by simply registering the webservice to all the legal subjects corresponding to the underlying services.

To illustrate, suppose it is required to construct a composite webservice based on two services *service1* and *service2*. If *service1* requires read-only access to legal subject *s1* and *service2* requires update access to legal subject *s2*, the following registration entry can be set up for the new webservice: <*ws*, *s1*, *read>, <*ws*, *s2*, *update>.

This offers a simple and powerful way to establish the access rules for new services, as compared to the other access control models.

10 Conclusion and Future Directions

In this paper, we have examined the key requirements of the eGovernment and how the webservices technologies can be applied in the government domain. This reveals that most of the issues faced by eGovernment are about integration and interconnectivity. Webservices technologies, which is a tool for universal interoperability, is therefore a logical solution to most eGovernment problems. We have also outlined a webservices based eGovernment infrastructure with four different webservice models. To incorporate these models in the eGovernment context, an eGovernment architect will be able to make the following important design decisions:

- decide the functions to be exposed as eGovernment webservices (the system component model);
- decide the right access means to each eGovernment webservice (the system access model);
- establish the right access policy for each eGovernment webservice and the appropriate security mechanisms to implement it (the security model);
- establish appropriate mechanism for the eGovernment webservice transactions (the transaction model).

We have thus provided the essential reference models for a webservice eGovernment framework to demonstrate how the webservices technologies can be applied in the government domain.

In addition, we have argued that a security infrastructure implemented at the global level is essential to protect a webservices based eGovernment system. The detailed requirements of such a mechanism have been examined. Lastly, we have illustrated how the security infrastructure can be implemented by proposing a mechanism that can address all of the requirements.

The application of webservices technologies and in particular in the government domain is a relatively new subject. There are thus a lot of areas for further research. We would like to close this paper by pointing out a few of them.

Firstly, on the eGovernment webservice component model, researches can be done to provide more details to the model. What we have done is largely on the initial steps to convert the legacy government systems into a service based structure and thus the focus should be on the extraction of the existing business functions into webservice components. However, once this step is largely completed, additional webservices can be implemented to provide new business services. Alternatively, new webservices can be created on top of existing ones to provide macro-services across government departments. Work is thus desirable to see how this can be best done. Some of the research topics are: Is there an optimal granularity of the functionalities to be provided in an eGovernment webservice? Is it desirable and how to employ a tiered or layered based component model to the eGovernment webservice?

For the transaction model, we believe that once the webservice transaction standard is becoming mature, an infrastructure framework to support eGovernment webservice transaction will be required. Research in this area may result in the formulation of either a general or eGovernment specific framework to implement webservices based workflow.

In this paper, we have identified the legislation based access policy as an important source of eGovernment access policy and we have proposed a simple structure to accommodate the corresponding access rights. We have not attempted to prove that this is the best approach to capture the legislation based rights. It remains an open question whether a more supplicated structure (eg a tree or directed graph structure) can capture the legal subject more effectively, which result in a better access control mechanism. In particular, whether such an alternative approach can make it more feasible to capture the policy into the system automatically.

We have argued throughout the paper that interoperability is a key eGovernment issue which can be addressed with a webservices based structure. However, a number of recent events indicate a trend of much more and closer cooperation among governments of different countries. These include, for example, the formation of the European Union and the joint effort to fight against terrorism and more recently, the SARS disease. We speculate that this may in time call for more integration of the systems between governments to achieve the necessary coordination of the efforts. It is thus a challenging but interesting question to see whether and how the eGovernment webservices framework can be extended to accommodate the level interoperability beyond a single government.

References

1. J. L Ambite, Y. Arens, E. Hovy, A. Philpot, L. Gravano, V. Hatzivassiloglou and J.Klavans (2001), "*Simplifying data access: the Energy Data Collection project*", IEEE Computer, Vol 34 Issue 2, pp. 47-54.
2. F. Arcieri, E. Cappadizzi, E. Nardelli and M. Talamo (2001), "*SIM: a working example of an e-government service infrastructure for mountain communities*", Proceedings of 12th International Workshop on Database and Expert System Applications, pp. 407-411.
3. B. Atkinson, G. D. Libera, S. Hada, M. Hondo, P. H. Baker, J. Klein, B. LaMacchia, P. Leach, J. Manferdelli, H. Maruyama, A. Nadalin, N. Nagaratnam, H. Prafullchandra, J. Shewchuk and D.Simon (2002), "*Web Services Security (WS-Security)*", IBM developerWorks: WS-Security Specification Version 1.0, http://www.ibm.com/developerworks/library/ws-secure/
4. E. Bertino, A. Perego, E. Farrari, N. R. Adam and V. Atluri (2002), "*DLAM: An Access Control Model for Digital Libraries*", Proceedings of the Second International Conference on Information Security, Shanghai, pp 149-155.
5. T. Boubez, M. Hondo, C. Kurt, J. Rodriguez and D. Rogers (2002), "*UDDI Data Structure Reference V1.0*", UDDI Published Specification, 28 June 2002, http://www.uddi.org/pubs/DataStructure-V1.00-Published-20020628.pdf
6. A. Bouguettaya, M. Ouzzami, B. Medjahed and J. Cameron (2001), "*Managing government databases*", IEEE Computer, Vol 34 Issue 2, pp. 56-64.
7. S. Burbeck (2000), "*The Tao of e-business services: The evolution of Web applications into service-oriented components with Web services*", IBM

developerWorks: Web services, October 2000, http://www-106.ibm.com/developerworks/library/ws-tao/index.html

8. D. Chappell and T. Jewell (2002), "*Java Web Services*", Sebastopol, CA: O'Reilly.

9. G. Coulouris, J Dollimore and T Kindberg (2001), *"Distributed Systems: Concepts and Design (Third edition)"*, Essex, England:Addison Wesley, Chapter 3 (Networking and Internetworking)

10. B. Eckel (2000), *"Thinking in Java, 2nd Edition"*, Upper Saddle River, NJ:Prentice Hall, Chapter 1 (Introduction to Objects).

11. The Economist (2000), "*No Gain Without Pain*", The Economist, 24June2000, http://www.economist.com/displayStory.cfm?Story_ID=80764

12. J. Feller (2002), "*IBM Web Services Toolkit – A showcase for emerging web services technologies*", Web services by IBM, http://www-3.ibm.com/software/solutions/webservices/wstk-info.html

13. D.F. Ferraiolo and D.R. Kuhn (1992), "*Role Based Access Control*", 15th National Computer Security Conference.

14. A. Gore (1993), "*1993 Report: From Red Tape to Results: Creating a Government that Works Better and Costs Less*", First NPR report presented by the Vice President Albert Gore to President Clinton, Washington, Chapter 1 (Executive Summary), http://govinfo.library.unt.edu/npr/library/nprrpt/annrpt/redtpe93/index.html

15. S Graham, S. Simeonov, T Boubez, D. Davis, G. Daniels, Y. Nakamura and R. Neyama (2001), "*Building Web Services with Java: Making Sense of XML, SOAP, WSDL and UDDI*", Indianapolis, IN: SAMS.

developerWorks: Web services, October 2000, http://www-106.ibm.com/developerworks/library/ws-tao/index.html

8. D. Chappell and T. Jewell (2002), "*Java Web Services*", Sebastopol, CA: O'Reilly.

9. G. Coulouris, J Dollimore and T Kindberg (2001), *"Distributed Systems: Concepts and Design (Third edition)"*, Essex, England:Addison Wesley, Chapter 3 (Networking and Internetworking)

10. B. Eckel (2000), *"Thinking in Java, 2nd Edition"*, Upper Saddle River, NJ:Prentice Hall, Chapter 1 (Introduction to Objects).

11. The Economist (2000), "*No Gain Without Pain*", The Economist, 24June2000, http://www.economist.com/displayStory.cfm?Story_ID=80764

12. J. Feller (2002), "*IBM Web Services Toolkit – A showcase for emerging web services technologies*", Web services by IBM, http://www-3.ibm.com/software/solutions/webservices/wstk-info.html

13. D.F. Ferraiolo and D.R. Kuhn (1992), "*Role Based Access Control*", 15th National Computer Security Conference.

14. A. Gore (1993), "*1993 Report: From Red Tape to Results: Creating a Government that Works Better and Costs Less*", First NPR report presented by the Vice President Albert Gore to President Clinton, Washington, Chapter 1 (Executive Summary), http://govinfo.library.unt.edu/npr/library/nprrpt/annrpt/redtpe93/index.html

15. S Graham, S. Simeonov, T Boubez, D. Davis, G. Daniels, Y. Nakamura and R. Neyama (2001), "*Building Web Services with Java: Making Sense of XML, SOAP, WSDL and UDDI*", Indianapolis, IN: SAMS.

16. T. Härder and A. Reuter (1983), *"Principles of Transaction-Oriented Database Recovery"*, Computer Surveys, Vol 15, No 4, pp 287-317.

17. S. C. Henderson and C. A. Snyder (1999), *"Personal information privacy: implications for MIS managers"*, Information & Management, Vol 36 Issue 4, pp 213 – 220.

18. HKSAR (1996), "*PERSONAL DATA (PRIVACY) ORDINANCE*", Laws of Hong Kong, Chapter 486, s 2 (Interpretation).

19. HKSAR ITSD Hong Kong Special Administrative Region Government Information Technology Services Department (2001), "*General Policy*", Building A Digitally Inclusive Society, Chapter 3, p 22.

20. P. Jackson and N Curthoys (2001), "*E-government: developments in the US and UK*", Proceedings of 12th International Workshop on Database and Expert System Applications, pp. 334-342.

21. J. Joshi, A. Ghafoor, W. G. Aref, E. H. Spafford (2001), "*Security and Privacy challenges of a Digital Government*", Advances in digital government : technology, human factors, and policy, Norwell, MA:Kluwer Academic Publishers, Chapter 7, pp. 121-136.

22. H. F. Korth, E. Levy and A. Silberschatz (1990), *"A Formal Approach to Recovery by Compensating Transactions"*, Proceedings of 16th International Conference on Very Large Databases.

23. B. Lampson (1969), "*On reliable and extendible operating systems*", Proceedings 2nd NATO Conference on Techniques in Software Engineering, Rome. Reprinted in "*The Fourth Generation*", Infotech State of the Art Report 1 (1971), p 421.

24. B. Lampson (1971), "*Protection*", Proceedings 5th Princeton Conference on Information Sciences and Systems, Princeton, p.437. Reprinted in ACM Operating Systems Rev. 8, 1 (Jan. 1974), pp 18-24.

25. D. S. Linthicum (2000), *"Enterprise Application Integration"*, Reading, MA:Addison-Wesley.

26. C. V. Lopes and W. L. Hursch (1995), "*Separation of Concerns*", College of Computer Science, Northeastern University, Boston.

27. M. Mecella and C. Batini (2001), *"Enabling Italian e-government through a cooperative architecture"*, IEEE Computer, Vol 34 Issue 2, pp. 40-45.

28. J. Mechling (1994), "*Reengineering government: Is there a 'there' there?*" Public Productivity & Management Review, Thousand Oaks; Vol 18, Iss. 2, pp. 189-197.

29. MSDN Library (2001), "*XML Web Services*", http://web.archive.org/web/20011129064207/http://msdn.microsoft.com/library/default.asp?url=/nhp/Default.asp?contentid=28000442

30. MSDN Library – BizTalk Server (2002), "*XML Web Services*", Microsoft Biztalk Server and Visual Studio .NET, http://msdn.microsoft.com/library/default.asp?url=/library/en-us/bts02kit/htm/bts_netsdk_gettingstarted_hgsr.asp

31. J. M. Myerson (2002), "*Web Service Architectures - How they stack up*", Tect, http://www.webservicesarchitect.com/content/articles/webservicesarchitectures.pdf

32. OASIS Organization for the Advancement of Structured Information Systems (2002),"*Business Transaction Protocol*", Oasis BTP Draft Specification 0.9.6.2,

http://www.oasis-open.org/committees/business-transactions/documents/2002-05-16.BTP_draft_0.9.6.2.pdf

33. OECD Organisation for Economic Co-operation and Development (2002), "*Annex Table 26. General government total outlays*", OECD Economic Outlook, 71, p. 232.

34. W. L. Oellermann (2001), *"Architecting Web Services"*, Berkeley, CA:Apress, pp 262-270(Application Authentication).

35. H Schorr and S J Stolfo (1997), "*Towards the Digital Government of the 21st Century*", Workshop on Research and Development Opportunities in Federal Information Services report, June 1997, http://www.isi.edu/nsf/final.html

36. C. Scott, P. Wolfe and M. Erwin (1999), "*Virtual Private Networks, Second Edition*", Sebastopol, CA: O'Reilly.

37. Sun Microsystems (2002), "*Delivering Services on Demand*", Sun ONE Architecture Guide, Chapter 1, http://wwws.sun.com/software/sunone/docs/arch/chapter1.pdf

38. Taylor Nelson Sofres (2001), "*Government Online – an international perspective*", 2001 Benchmarking Research Study.

39. R. Thomas & R. Sandhu (1997), "*Task-based Authorization Controls (TBAC): A Family of Models for Active and Enterprise-oriented Authorization Management*", 11th IFIP Working Conference on Database Security.

40. R. Traunmuller and M. Wimmer (2000), "*Processes-collaboration-norms-knowledge: signposts for administrative application development*", Proceedings of 11th International Workshop on Database and Expert System Applications, pp. 1141-1145.

41. F. Virili (2001), "*The Italian e-government action plan: from gaining efficiency to rethinking government*", Proceedings of 12th International Workshop on Database and Expert System Applications, pp. 329-333.

42. W3C (2000), "*Simple Object Access Protocol (SOAP) 1.1*", W3C Note, 08 May 2000, http://www.w3.org/TR/SOAP/

43. W3C (2001), "*Web Services Description Language (WSDL) 1.1*", W3C Note, 15 March 2001, http://www.w3.org/TR/wsdl/

44. Wil (2000), *"Loosely coupled interorganizational workflows: modeling and analyzing workflows crossing organizational boundaries"*, Information & Management, Vol 37 Issue 2, pp 67 – 75.

45. M. Wimmer and B. von Bredow (2001), "*E-government: aspects of security on different layers*", Proceedings of 12th International Workshop on Database and Expert System Applications, pp. 350-355.

46. R. Yahalom, B. Klein and T. Beth (1993), "*Trust Relationships in Secure Systems – A Distributed Authentication Perspective*", Proceedings of the 1993 IEEE Symposium on Research in Security and Privacy, pp 150-164.